April 1983 75p

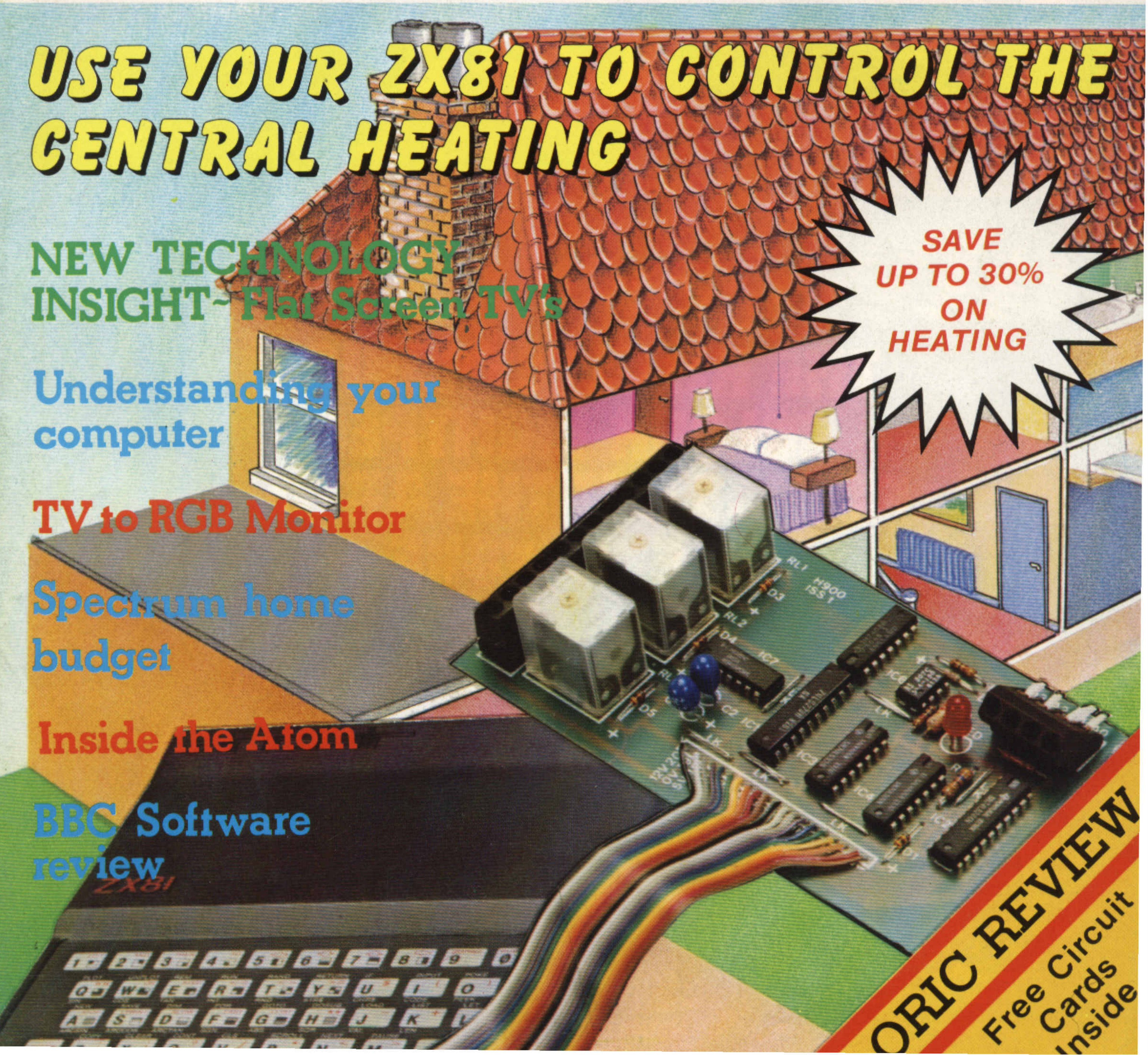
9 PRINTER 23 BUDGET ACCOUNT 28 DIG. ELEC.
34 37 41 KITCOMPUTER 40 ORIC 53 ERROR 52 WORD REF 55 APPLIC-ATION PROC

Electronics & computing

MONTHLY

62 RGB
65 66 67 74 LEADS 77 WORD PROC REF 78 PIRATE REF
94 AD-ROBOT PARTS

Britain's **First** Electronics & Computer **Applications** Magazine



USE YOUR ZX81 TO CONTROL THE CENTRAL HEATING

SAVE UP TO 30% ON HEATING

NEW TECHNOLOGY INSIGHT~Flat Screen TV's

Understanding your computer

TV to RGB Monitor

Spectrum home budget

Inside the Atom

BBC Software review

ORIC REVIEW

Free Circuit Cards Inside

ELECTRONICS & COMPUTING MONTHLY
40/42 Oxford Street
Daventry
Northampton
NN11 4AD
(03272) 71841/71702

Publisher/Editor
DAVID RAVEN

Advertising Manager
Claire Fullerton

Deputy Advertising Manager
Alun Evans

Production
TERRY GIBBINS

Editorial Assistant
Ann Houghton

Office Manager
Ruby Jordon

Distribution
Circulation Department
EMAP National Publications Ltd
Bretton Court
Bretton
Peterborough
0733-264666

Published By
EMAP Business & Computer Publications
Bushfield House
Orton Centre
Peterborough
0733-237111

Publishing Director
Peter Peskett

Production By
Time Graphics
Northampton

Printed By
L.S.G. Lincoln

Vol. 3 Issue 4

IN THIS ISSUE

Electronics & Computing Monthly is normally published on the 13th day of each month

COMMENT

The last three years have seen several exciting stages in the U.K. development of microcomputers, the first of which was the low cost micro itself. With prices crashing down for excellent machines, the cost of software is becoming proportionally high.

One new technique, which could help to substantially reduce the cost of software, is electronic publishing in the form of the recently announced Micronet.

Micronet 800 is a new service operated in conjuction with Prestel which provides micro users with access to hundreds of computer programs for most of the popular microcomputers on the market. Britain is the first country in the world to operate such a system and this will clearly have a major effect on the already advanced development of software in this country. The advantages of this system over the traditional methods of typing in huge listings, which may or may not work after you've finished, are all too easy to see. But, in addition, Micronet could prove to be the first stage in a major move away from traditional forms of publishing and the beginning of an electronic mail service.

The lower cost of obtaining software, plus the ease with which subscribers can simply call up a particular program selected from a huge choice being incorporated for a wide range of different machine, makes Micronet an attractive enough idea. Add to this however, the full range of services offered by Prestel, which Micronet subscribers automatically benefit from, and it changes from desirable to necessary. The technique used to supply software to the end user is via a modem linked to the domestic telephone, and to the subscribers own computer. The micro keyboard is used to talk to Micronet, and software is charged so much per program down loaded, this is apart from the 100 pieces which will be available free for each model of computer incorporated on the Micronet system. The cost is currently £52, which includes a modem, for one year's subscription. Readers who want more information should ring 01 359 3699.

PUBLISHER

ELECTRONICS & COMPUTING DESIGN:

Circuit Designers — professional or otherwise — lend me your ears . . . If you have designed a novel or interesting circuit which could benefit your fellow electronics and computing enthusiasts — and you like CASH . . . then don't throw away that used table napkin or empty fag-packet which your latest design is on send it straight to:

THE EDITOR
E. & C. DESIGN,
40/42 OXFORD ST.,
DAVENTRY.
NORTHANTS.

we will in turn — after publishing your master piece — rush back to you £25.00 in any legal currency — but preferably by cheque.

DO IT NOW AND TURN YOUR: "THINKS INTO DRINKS!!!

NEXT MONTH

NEXT MONTH

Electronics & Computing Monthly

SPECTRUM RS232 Printer Interface

Build an RS232 input output port for your Spectrum. This will enable users to connect up to a wide range of printers and other accessories, in addition to inter-machine communication.

INTERFACING THE DRAGON 32

Construct a simple parallel interface for the Dragon.

PLUS: Jupiter Ace Electronic Scrambler, Model a Rotary Engine on The ZX81 and bring Power Control To Your Micro.

Regular features on the BBC Micro and other popular models every month in

BRITAIN'S FIRST ELECTRONICS AND COMPUTER APPLICATIONS MAGAZINE

On Sale 13th APRIL

LETTERS

Dear Sir,

As a regular reader of your magazine for over a year now. I am writing to echo the sentiments expressed by (R. Margree of Drayton). "Why must it be devoted almost entirely to Sinclair users?"

Your reply to the Dragon enthusiast gave the reason that the machine has not been around very long, but I wish to remind you that the Atari 400/800 has been around for a while now, and it seems strange to me that Britain's FIRST (and I might add BEST) Electronics and Computer APPLICATIONS MAGAZINE has neglected this machine which, with it's onboard slave chips, – POKEY and P.I.A. – coupled with the Atari 850 Interface module, (with it's own microprocessor), provide for a very flexible interface system.

The "850", with it's four RS-232 serial ports and one paralleled port, has a user specific baud rate ranging from 45.5 to 9600, and is easily controlled using Atari Basic's X10 # command plus all the more usual I/O commands.

I hope to see some Atari programs in the future, and would like to thank you for a fascinating read around the 15th of each month.

I should also like to say that your Sinclair programs provide me with many useful ideas for my machine.

Yours faithfully,

Jeffrey Best
83 Chippers Road,
Worthing,
W. Sussex.

Dear Sirs,

With reference to E & CM January 83 issue and (A. Hawkins of Swansea) comment about articles being too technical.

I wish to build an A to D convertor for a Spectrum and have over the last few months studied various technical notes. On seeing the magazine in the School Library I thought that at last I had found the information but alas no! only more technical information.

Most of us learning about computers need at least circuit diagrams, perhaps ever leading by the hand, not a discussion article.

John Taylor does appear to know what he is talking about but much more explanation of how to do it including a final circuit diagram and suggested layout please.

Articles and information on 'Control' using computers are needed for schools please help by ensuring that the likes of us can follow, build and understand them.

Yours faithfully,

B. Hoggarth (Teacher)
Latimer School
Barton Seagrave
Kettering

Dear Sir,

May I ask for the hospitality of your columns to pass on to your readers news of two user groups which are based in Nottingham here.

First: the TRS-80/Genie Users Group of Nottingham, which has been running as a thriving society here for nearly three years will be very happy to meet users in the are at Wilford Moderns Rugby Club House at any forthcoming meeting. Meetings are generally held on the first and third Wednesday in each month from 7.30 p.m. A Club magazine, LPRINT is published fairly regularly. For further information please telephone or send an SAE to the undersigned or the Club Chairman, Marc Leduc, Marcos Software, 30 Waterloo Road, Beeston, Nottingham 225165.

Second: the National Colour Genie User's Group has now been formed, with Marc Leduc as Chairman. Full particulars and a sample copy of the club magazine will be sent to anyone sending me a largish SAE.

With many thanks.

Geoffrey Hillier,
Editor, LPRINT
5a Gregory Street,
Lenton, Nottingham NG7 2LR
Tel: 783938

Dear Sir,

Would you please publish the details of the Grimsby Computer Club when possible in your magazine.

Meetings are held on alternate Mondays at St. James Hall, Grimsby.

The club is not computer specific.

Membership is £8.00 per year with reductions for schoolchildren, OAPs and father/son joint memberships. A periodical newsletter is published.

Regular features are Family nights, Basic computer courses and Business Familiarisation courses.

We also hope to hold our second Computerfair this year after the terrific response to our first attempt in December last.

Any further details may be obtained from Jenson Lee (below) or myself on Grimsby 49248 after 6 p.m.

Yours faithfully,

Ian Fell (Publicity Officer)
Grimsby Computer Club
29 Park View
Cleethorpes
Tel: (Day) 42559

Dear Sir,

In your article "Spectrum Eprom Programmer" (John Williams Feb 83), the author has interfaced the programmer with the Spectrum Z80A using a "slow memory" technique. Memory write cycles to the EPROM being programmed are extended by inserting Wait states; in the published design memory write cycles are 50 mS long which is beyond the scope of the interfacing technique.

The Z80A does not refresh during wait states, and consequently for each write to the EPROM the Spectrum's Dynamic RAM is deprived of refreash for 50m S D – RAM used in the spectrum is 4116 and 4132 type which has a refresh requirement of one complete cycle every 4m S.

Readers using the programmer may find that RAM contents (i.e. program and system code) become corrupt because of insufficient refresh.

To minimise this affect I would suggest . . .

1 That sufficient code is executed between successive POKES to allow 128 M1 cycles to be executed e.g.

```
10 LET EPROM = 0 : LET BYTE
   = 157
20 POKE EPROM, BYTE
30 FOR WAIT = 1 to 10 : NEXT
   WAIT : REM WAIT 128 M1
   cycles.
40 POKE EPROM + 1 , BYTE –
   79
50 . . . . . . . . etc.
```

2 That the programmer is used as quickly after switch-on as possible before the Spectrum's D-RAM has achieved full working temperature.

Yours faithfully,

D. Davidson
15 Maple Close
Oxford

P.S. THANKS FOR AN APPLICATION ORIENTATED MAGAZINE.

New Low Priced Colour Computer

Texet, who were the first UK company to produce a pockket calculator in 1971, have achieved another major breakthrough with the introduction of their new easy to use TX8000 colour computer, which will be available from March, and is aimed at the Home Computer Market. This exciting new machine, at £98, represents a substantial price breakthrough in micro-processing technology, being manufactured by one of the foremost manufacturers of microchip leisure products in the world.

Texet's highly experienced management team in consumer electronics have brought together new information, new technology, new developments and new products in fields of consumer electronics and entertainment.

Texet will be concentrating on distributing the TX8000 through leading store groups such as John Menzies, Debenhams, Asda, Makro, Fine Fare, House of Frazer, Lewis's and Woolworths, although a proportion of stock will be reserved for smaller retailers specialising in computers in order to cover the rapidly expanding market.

The eight colour computer (green, yellow, blue, red, buff, cyan, magenta, orange) is expandable from 8K to 64K RAM with a memory expansion module at £52, which will give more experienced programmers greater scope to use their acquired skills. Microsoft basic, the univesal programming language, is used. The full size, moving key rubber keyboard has 45 automatic repeat keys, 16 pre-defined single character graphic keys, single keyword or standard typing entry, and a 'beep' sound on key depression. The cassette interface card operates at a baud rate of 600 bps, and connects to any standard audio cassette recoreder. A memory expansion bus and a peripheral expansion bus are built right into the machine. Visual output can be either to a television set or to a monitor and the power supply is a single 9 volts DC, the adaptor being included.

The TX8000 is an extremely compact machine, measuring 29 x 16 x 5cm, which is ideal for the home user, and the expansion module measures 14 x 8 x 3.5cm.

Texet will be supplying a fantastic range of software to accompany the launch of their exciting new machine, ranging from Home Finance to Video Games. Texet have acquired a wide selection of joysticks available from the very competitive price of £7.95 – a single button model going through dual buttons, left and right buttons, to a cordless remote control model at £59.95 per pair. Also available from Texet will be a printer for £129, monitors, cassette players, computer cassette tapes, light pens and a printer interface, which should satisfy all the needs of the Home Computer user.

A Computer Programming Weekend by Ann Houghton

This residential course, which costs £5 including all food, accommodation and VAT, started on Friday evening with sherry in the Military Bar, and general introduction between the 20 people on the course, and the lecturer, Harry Siddall.

The serious business started after dinner with setting up the computers and switching them on. ZX81's were used with two people to each machine.

This course was aimed at the conplete beginner to programming, so started with explanations about direct commands, progressing on to simple programs. Several members of the course were not absolute beginners, and were encouraged to go ahead and try more complicated programs. RAM packs were available for those who needed extra 'K'.

At the end of the evening's instruction several people stayed behind to carry on working. Harry was only too pleased to stay also, to help and advise. Another group made a mad dash to the bar, and were reassured to find that closing time was non-existant where residents were concerned.

Saturday morning saw one or two bleary eyes faces at the breakfast table but a proper English breakfast soon revived most people. Instruction on "programming with style", including loops, RND, plotting, INT etc. continued through the morning, with a break for coffee and biscuits.

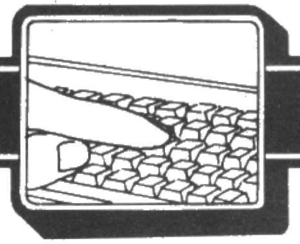
After an excellent buffet lunch there was more instruction until our free time when we were able to explore local places of interest. Some, however, chose to stay and work for a while.

After dinner there was more instruction, followed by discussions and comparing of ideas in the bar.

Sunday followed a similar pattern to Saturday with the course finishing at 5 p.m.

The entire weekend was thoroughly enjoyable, both socially and instructionally.

Details from Gainsborough House Hotel. Tel: 0562 754041.



New TI Home Computer

A new computer console believed to be the first 16-bit computer for around £75 has been announced by Texas Instruments. Unlike most computers in this price range, the TI-99/2 basic computer can use software on solid state cartridges as well as on cassettes. In addition, Texas Instruments is introducing new low cost peripherals and software for the TI-99/2 that will also work with the TI-99/4A.

"The TI-99/2 is designed to allow computer novices to learn to program a computer in TI BASIC and BASIC-supported assembly language", said William Turner, President of the Consumer Group.

"The combination of the £75 computer and the new low-cost peripherals make this the lowest cost computer system on the market. The TI-99/2 will be targeted primarily at the technical enthusiast, engineer or student in the home. Additionally, we expect the computer to be purchased as the first computer in the home for those who are just beginning their experience with a computer system, or as-a second computer in the home after the purchase of a TI-99/4A family computer."

The TI-99/2 console has an elastomeric typewriter-like keyboard with 48 raised moving keys in a staggered QUERTY arrangement similar to the TI-99/4A. The computer has 4.2K bytes of built-in random access memory (RAM), of which 4K Bytes is user accessible, and can be expanded to a total of 36.2K bytes of RAM.

Most peripherals for the new system will plug into a Hex-bus* peripheral-interface connector in the rear of the console. The Hex-bus port allows users to connect any peripheral developed for TI's Compact Computer family. Currently, these consist of the RS232 interface, HX-3000, the Wafertape* digital tape drive unit, HX-2000, and the HX-1000 four colour printer/plotter.

Two Solid State Software* cartridges, "Learn to Program" and "Learn to Program BASIC", will be available initially for the unit. Other cartridges will be available later.

Twenty software programs will be available on cassettes at the initial launch. Educational programs include: Picomath-80, Math I and II, Statistics I and II, Sunrise Time, Datetimer and Civic Engineering. Programs for personal management are: Household formulas, chequebook manager, purchase decisions and general finance. Entertainment cassettes include: Lunar Landing Bioplot, The Minotaur, TI Trek, and Mind Games I, II, III and IV. These programs, as well as user-written programs, can also be run on the TI-99/4A family computer.

The TI-99/2 features monochrome display capability and contains a built-in RF modulator to allow connection to any television. A cassette interface cable is also included to interface directly to the new TI program recorder or many ordinary cassette tape players. In addition, the TI-99/2 comes with an ac adapter, a user's manual and a demonstration cassette.

The TI-99/2 measures 24cm x 26cm x 35cm.

Availability in Europe is planned for the third quarter of this year.

*Registered Trademarks.

Cheetah RAM Packs

The Cheetah Ram Packs come in two different versions that simply plug into the user port at the rear of the Sinclair ZX 81 Personal Computer and are fully compatible with other accessories.

Both units are professionally cased in a custom made unit, fully tested and guaranteed.

The connector is gold plated and coated with a special chemical to ensure extra long life.

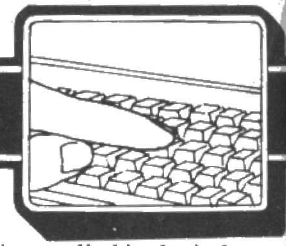
The case, which is identical for both versions, is designed to fit the computer snugly, therefore, ensuring that the program is not lost through wobble.

For further information contact: Cheetah Marketing Ltd., 359 The Strand.

"ULTIMATE 16 BIT MACHINE"

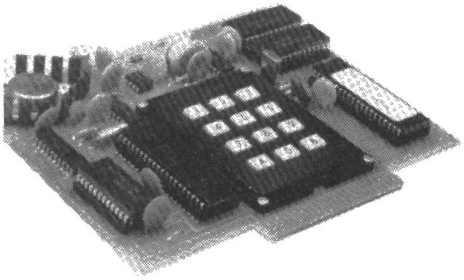
Almarc Data Systems of Nottingham have announced their new Series 16 micro computer. The Series 16 is a high performance machine based around the Intel 8MHz 8086 processor and also supports the 8087 maths processor and the 8089 I/O chip. Standard Ram supplied is 128K and this can be extended to a full 1 megabyte. Five disk options are available from 1.6 megabytes to 20 megabytes and an expansion unit allows for a further 40 megabytes and a tape drive to be added.

Alan Hood, Almarc's Chairman commented "We are very excited about this machine because it provides high performance but utilises the Intel chip thus allowing access to vast amounts of software that will no doubt become available following I.B.M's lead in this area. We are using the S100 bus and this allows our Series 8 machines to be upgraded to Series 16's and also make a great number of add on boards available. When asked how the Series 16 fits into their current range Hood said "Along with out major supplier, Vector Graphic, we have always been at the higher end of the market and we see this machine complementing the existing Vector 4 range of equipment by offering a higher performance option. At the moment we offer CP/M 86 as the resident operating system but in the near future we shall also offer MSDOS and MP/M 86. Other operating systems are also being looked into. We see the Series 16 as the ultimate 16 bit machine, providing



superb performance coupled with maximum hardware and software flexibility".

Prices start at under £3000.00, further details are available from:- Almarc Data Systems Ltd., Great Freeman Street, Nottingham, NG3 1FR. Tel: 0602 52657.



Speech-synthesis boards provide evaluation and standard vocabulary

Two speech-synthesis microcomputer boards for the National Semiconductor Digitalker system are now available from Hi-Tek Distribution. The DT1000 is a speech-synthesis evaluation board which enables potential users to understand the operation and application of the Digitalker chip set in an end product, while the DT1050 is a standard Digitalker kit encoded with a vocabulary of 137 words, two tones and five different silence durations.

The DT1000 contains all the components required to produce speech on demand, including the processor chip, memory, filter, amplifier, keyboard and a microcontroller pre-programmed with stored data. The only external components required are a single 7-11V power supply and a loudspeaker.

The two read-only memories on the board contain a brief introductory phrase, 138 separate and individual 'words' (including numbers, letters, nouns, verbs and tones), and five different individual silence durations. A program known as COPS allows the user to output each word sequentially, repeat any desired word, build and store short phrases, output and edit pre-programmed phrases, play a simple interactive game, and output hex equivalent decimal number inputs.

The DT1050 standard vocabulary set is chosen to be applicable to many products and markets. A highly intelligible male voice is used, with the natural inflection and emphasis of original speech. The words and tones are assigned discrete addresses, making it possible to output single words or words joined into phrases or sentences. The system can be easily interfaced with a range of microprocessors, and on-chip switch debounce circuitry is provided for interfacing to manual switches independently or a microprocessor.

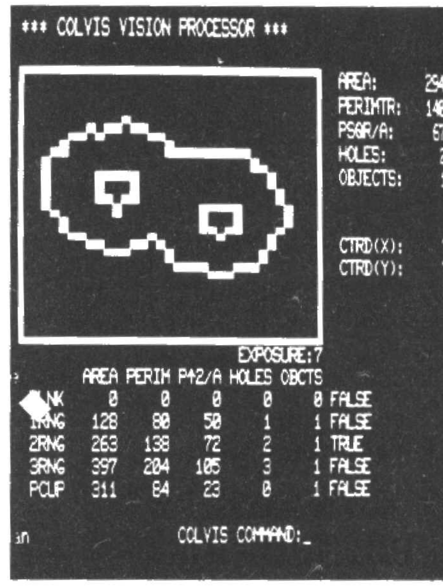
Hi-Tek Distribution Limited, Trafalgar Way, Bar Hill, Cambridge, CB3 8SQ.

Note to Editors: For further information please contact: Dave Greenfield, Hi-Tek (Tel: 0954 81931) or, Peter Bush, Bush Steadman & Partners Ltd., 4 Cross Street, Saffron Walden, Essex, CB10 1EX. (Tel: 0799 23101/27240).

64K D-RAM uses latest technology for high performance

Available from Dialogue is Toshiba's new 64kbyte D-RAM (dynamic random-access memory), which uses the latest double-poly n-channel silicon-gate process technology and advanced circuit techniques to provide wide operating margins, both internally and to the system user.

The new TMM4164P is organised as 65 and 536 words by one bit, and operates from a single 5V power supply. Operating power is only 275mW maximum, and standby power is 27.5mW. The device has no connection on pin 1, and is supplied in the industry-standard 16-pin dual-inline plastic package.



British Company develops low-cost computer vision system

Colne Robotics of micro-robotic fame have now developed a low cost computer vision system.

The Colne Robotics system COLVIS, which permits a computer to see objects and remember their shapes, will be priced at £395.

It consists of a solid-state camera connected to a powerful micro-computer, which is capable of extracting and learning information from the image produced. This information comprises a range of parameters, such as area, perimeter and centre of gravity of the image. It is used to recognise the object in view, then deduce its position and orientation. The system can be used in conjunction with any micro-computer which has, or can be fitted with, an 8-bit bi-directional port.

As with Colne Robotics' existing robotic arm, the vision system is aimed at the educational market. A versatile teaching aid, equally at home in the University department or the classroom, it is also appropriate to the teaching carried out in Technical Colleges and by Industrial Training and Development organisations. It constitutes an invaluable low-cost peripheral to existing robotic arms.

For further information, contact: John Reekie or Alasdair Macmillan, Colne Robotics Co. Ltd., Beaufort Road, Off Richmond Road, East Twickenham, Middlesex, TW1 2PQ. Tel: 01 892 8197 or 8241.

Software for the Epson HX-20

The first of a range of office aides has been released, by Kuma making life easier for the business user of the HX-20.

DESK MASTER 1 unclutters the busy desk by removing the addlister calculator leaving more space for HX-20's etc. This microcassette based program causes the HX-20 to duplicate the functions of a normal printing calculator, e.g. add up a list of figures whilst printing them out. Many people in accounts departments and small businesses spend (seemingly) half their life doing just this.

In addition (no pun intended) comments, date and time of day can be easily added to the printer listing. Accuracy is set to 16 digits to aid the larger company and scientific calculations. Several options can be set and if required permanently stored within the program such as number of decimal places displayed and printed, floating decimal point, round off or truncate and accumulate results in a "Grand Total" memory register.

When not in use the display acts as a Clock/Calendar. Tim Moore, MD of Kuma, said "We found that the HX-20 was a very useful office tool once the software was there. This program is deceptively simple to use whilst performing a very useful everyday task".

For more details contact: Jon Day, Kuma Computers Ltd., 11 York Road, Maidenhead, Berks, SL6 1SQ. Tel: 0628 71778.

New Add-ons for ZX Users

Fuller Micro Systems, are launching yet another add-on for the Sinclair machines. The company, based in Sweeting Street, Liverpool, has sold more than 10,000 ZX 81 keyboard and keyboard case conversions since exhibiting a prototype at the Sinclair ZX Computer Fair in 1981.

Now they have incorporated a sound expansion unit in the casings, and plan to launch a redesigned keyboard incorporating a space bar facility on the ZX Spectrum kit.

ZX 81 users can buy the conversion system for £29.95, the keyboard and case kit for £24.95 or the keyboard kit only for £14.95.

The new redesigned keyboard with space bar and double size keys for shift and data entry will cost £39.95.

Several options are available for the 1983 range of Fuller equipment. For £54.90 customers can buy a Master Unit incorporating a Fuller Box, a sound amplifier, programmable sound generator, voice synthesiser and a joystick control.

The Fuller Orator, which translates typed in words from the keyboard into spoken words and sentences is £39.95, while a straight sound amplification unit costs £6.95.

The Fuller Box, which offers amplification, joystick control and a sound synthesiser costs £29.95 plus packing and postage and joysticks are £6.95 each.

"During 1983 we intend to expand our current dealership network considerably and also start to push the American market a lot harder", said Mr. Backhouse.

To date, Fuller have supplies conversion units to Europe, America, South America, Malaysia, the Middle East, even the Falkland Islands.

For more information contact: Mr. Roy Backhouse on 051 236 6109.

Torch Launch Prestel Computers

Torch Computers Ltd., have launched a new terminal for Prestel users in North America. The machine utilises fast down-loading software to dump data to disc quickly and accurately, giving real savings in the costs of call time. The software also allows time-delay operations for information transferral during cheap off-peak periods.

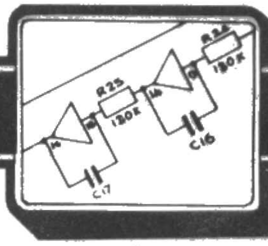
A 16-gate mainframe Prestel terminal will come on-line in Boston – the first micro-based VIEWDATA system ever installed for private users. Known as FLASHLIGHT, the central machine will use software developed in association with Metrotel to offer bulk update facilities and local VIEWDATA information.

The Prestel system will be built around Torch's communicating computer. This full colour machine offers interactive electronic mail as well as all the benefits of an advanced microcomputer. The standard machine has twin floppy disc drives, a high resolution tube, dual 8-bit processors and 96K of RAM. All Torch micros have an in-built modem, fully approved by the Federal Communications Commission in North America and British Telecom in the United Kingdom.

The system will employ the first ever micro-based VIEWDATA terminal, equipped with up to 100 modems, and available at a fraction of the cost of traditional systems.

Options to the system include hard disc machines of 10 or 20 megabytes, and the "SuperTorch" – a triple-processor machine with a 68000 chip and an additional 256K of RAM will shortly be available.

For further information: Torch Computers Limited, Abberley House, Great Shelford, Cambridge, CB2 5LQ, England. Telephone: Cambridge (0223) 841000.



THE FLAT CATHODE RAY TUBE

by M. A. White and M. S. White

Introduction

Ever since the first bulky Television Receiver appeared ('the Box') the desire for a flatter set has existed. The structure of the conventional cathode ray tube inhibits many further applications such as an aesthetically pleasing wall hung Television with a large screen or smaller flat screens for cars, caravans, boats, portable videophones and places where space is at a premium such as aircraft instrument panels. With the dramatic increase in personal computer ownership the potential consumer demand for flat screens must have reached a figure larger than the pioneers of flat screen CRT's could ever have envisaged.

The Pioneers

It was, in fact, an American advertisement which showed a wall hung Television that first inspired Dennis Gabor of Imperial College, London to think how he might make such a screen. At approximately the same time in Palo Alto, California William R. Aiken was pioneering research into flat cathode ray tubes. Both men had published scientific papers in the 1950's describing their flat screen designs and had developed working prototypes.

The first major design change necessary to produce a flat screen TV is in the position of the electron gun. One of the reasons for the conventional CRT's bulk is that the electron gun is both long and perpendicular to the screen. In order to produce the flatter model the electron gun must be moved parallel to the screen and still be able to produce a good picture.

Although both Aiken and the Gabor team made these design changes successfully, they were not at that time taken up by TV manufacturers due to their higher production costs. Gabor himself pointed out that the new flat screen would be more expensive. It may also have been that during the valve 'era' not much space would have been saved anyway because of the relatively large volume occupied by the valves and associated circuitry. However, with modern solid state circuit designs and the 'TV on a chip' goal the time seems ripe for the reappearance of the flat screen.

In more recent years several firms, including Texas Instruments in Dallas and RCA Laboratories in Princetown, N.J., have attempted to develop the flat CRT with varying degrees of success. The two most promising versions at present are those of Sinclair Research in England and the Sony Corporation in Japan.

How They Work

The Sinclair Version

In the Sinclair flat CRT the electron gun is placed on one side of the screen with its axis parallel to the screen giving rise to 'folded electron beam paths'. Horizontal and vertical scanning is made possible by two sets of electrostatic deflection plates in the gun assembly. The electron beam is bent towards the screen by a third set of plates formed by a front face transparent tin oxide electrode and the phosphor screen. (See Figure 1. reproduced by courtesy of Sinclair Research).

The picture is viewed through the front face electrode on the side of the phosphor screen where the electron beam lands and not through the phosphor as in the conventional CRT. Sinclair claim that this leads to a brightness gain of more than double the conventional tube brightness.

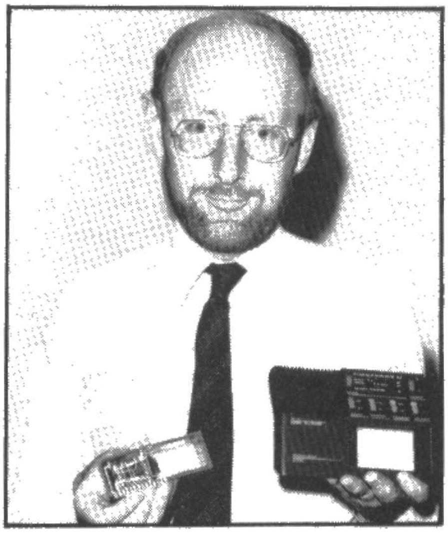


PHOTO CAPTION:

Clive Sinclair with one of the new Sinclair flat screen TV Tubes, and an early model prototype of the new Sinclair Microvision 2700, which incorporates the tube.

In the flat CRT the electron beam lands at an angle θ to the phosphor screen and the phosphor 'spot' increases by $1/\sin\theta$ times the beam diameter. In addition to this source of distortion the beam focussing position varies continuously during horizontal scanning. According to Sinclair these two effects cancel each other out during horizontal scanning 'so there is no significant change in spot size horizontally across the screen'. However, in the vertical direction the effect of scanning does not cancel out the increased spot size because the vertical deflection is slight. This leads to an eliptically shaped spot which is corrected for by a vertical magnifier – a flat Fresnel lens. Electronic

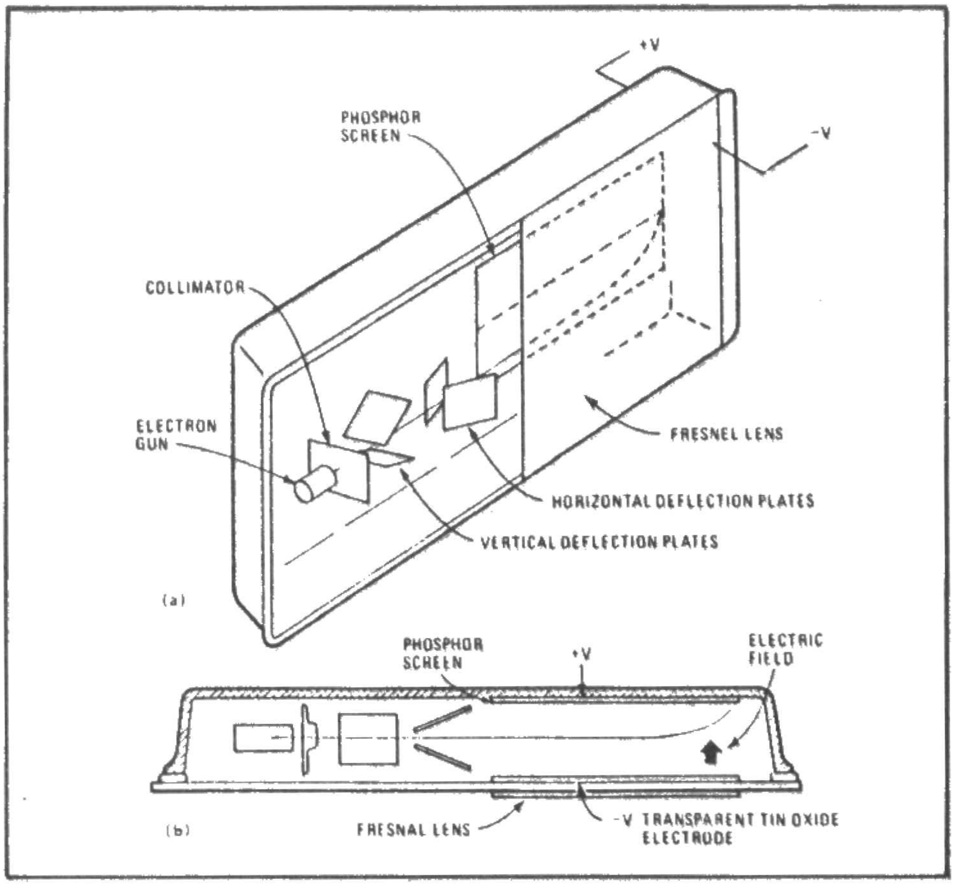


Figure 1 (a) and (b)
Sinclair Flat CRT schematic diagram.

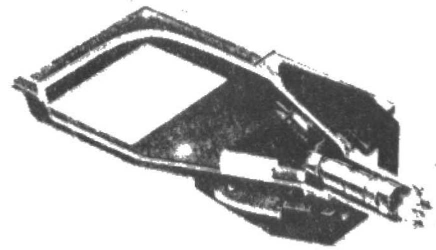


Figure 2
Schematic representation of the Sony CRT.

correction is also employed to overcome this distortion.

Power consumption is mainly due to the electron-filament with a quoted power consumption of 30mW.

The Sony Version

At the International Conference on Consumer Electronics in June 1982 Sony presented a paper on their flat cathode ray tube. A picture of this is shown in Figure (2) reproduced by courtesy of the Sony Corporation, Tokyo, Japan.

The electron gun is designed to be parallel to the phosphor screen so that, as in the Sinclair tube, it is necessary to bend the electron beam through 90°. The beam is horizontally deflected by meas of a **magnetic deflection yoke** attached to the outside of the CRT. Electrostatic deflection plates inside the tube achieve vertical deflection. A schematic diagram of this arrangement is shown in Figure 3 (a and b) (reproduced by courtesy of the Sony Corporation, Tokyo, Japan). Most power used by the CRT is in the horizontal deflection of the electron beam. A 14% reduction of power consumption was achieved by reducing the power requirements of the horizontal deflection system. This was done by shortening the distance between the magnetic poles of the deflection yoke. Even larger reductions in power consumption are claimed by using a high magnetic permeability ferrite material for the vertical deflection plates which double as pole pieces for the horizontal deflection yoke. Placed in the deflection magnetic field the pole pieces concentrate the magnetic flux generated by the horizontal deflection yoke and reduce the power requirement for horizontal deflection. Nevertheless, the horizontal deflection power consumption quoted is 150mW and the filament power is not given.

To overcome defocussing problems Sony employ a dynamic self-focussing method. This involves making the gun focussing potential vary as a function of the vertical scan potential. Additional circuitry is necessary to achieve this.

How They Are Made

In the Sinclair CRT two sheets of glass comprise the envelope. One of these, the image phosphor screen, is vacuum formed into a boat shape. The phosphor is coated by a dry electrostatic process onto aluminium electrode areas which have been first vacuum deposited on the glass. The other glass sheet, containing the transparent conducting repeller electrode, is silk screened with conductors. These form an edge connector on which electrodes are welded. The two glass parts are then bonded by a frit sealing process.

In the Sony model the front panel of plate glass is coated with the transparent conducting compound Indium Trioxide and forms part of the secondary deflection electrode. The glass envelope is carbon moulded at 1000°C, a method which is precise to ± 0.1mm. The part to which the deflection yoke is attached is ground down to 1.5mm thick to enable effective deflection of the electron beam by the yoke's magnetic field. The electrode's phosphor screen and glass sealing processes are similar to those described above for the Sinclair tube.

A New Approach

The Phillips Research Laboratories at Redhill, Surrey, have a radically different approach from the Sinclair and Sony designs. In the Phillips design the electron gun is mounted at the back and bottom of the viewing screen. The electron beam is projected towards the top of the screen and bent through 180° by means of a specially designed concave electrode held at cathode

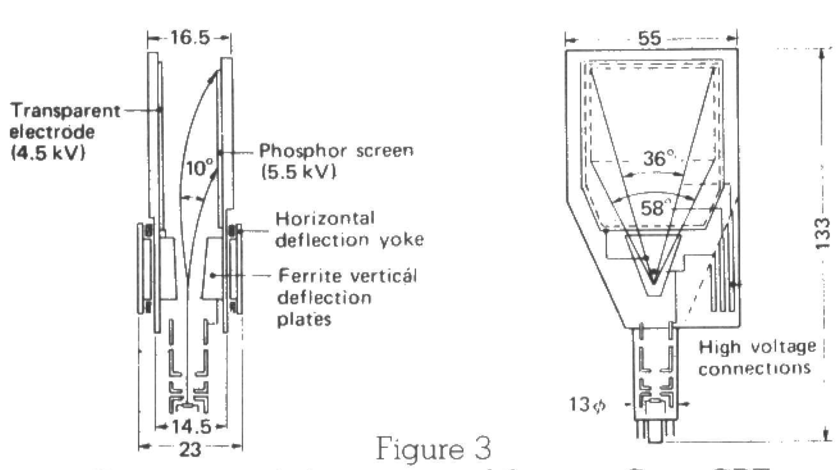


Figure 3
Structure and dimensions of the new Sony CRT. The deflection angles are indicated.

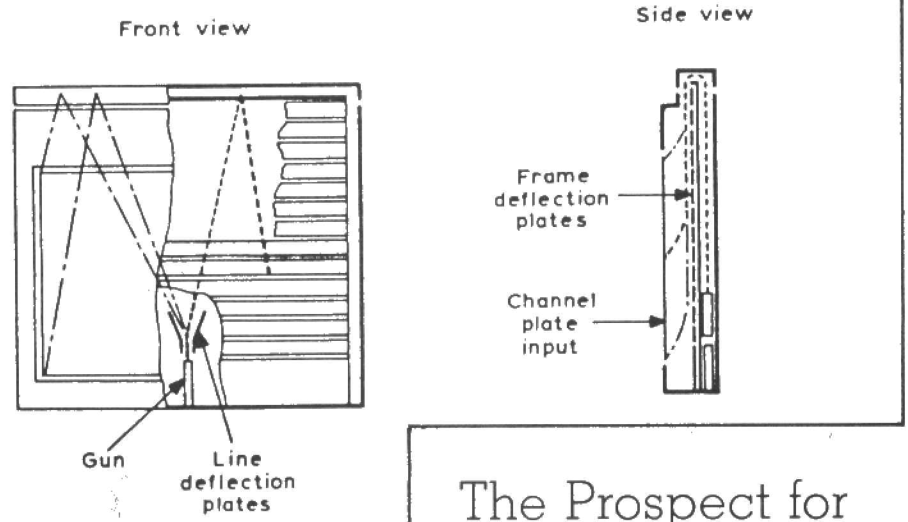


Figure 4
Phillips flat deflection system.

potential. The beam is then bent towards the screen and vertically deflected by a central plate which also separates the gun source from the screen, Figure 4. Line deflection is achieved by a pair of electrostatic deflector plates mounted within the electron gun assembly.

A particular difference in this design is that the screen consists of a specially designed channel electron multiplier. Phillips claim that this arrangement separates the functions of generating the light output and electron beam deflection. The primary modulated electron beam, which is typically one micro amp at 400V, only needs to scan a uniform raster at the entrance of the electron multiplier.

The Prospect for Colour

Colour in a flat screen television is as desirable as the flat screen itself. In all the writings on the subject it was implicit that colour would be a necessary option. Indeed, Professor Gabor's 1958 paper is entitled "A New Cathode Ray Tube for Monochrome and Colour Television" and the paper includes a substantial section devoted to colour control and its electron-optical realization. Aiken's flat CRT was also designed with colour in mind. Before they abandoned the whole project for commercial reasons, Texas Instruments had 'developed' an 8" x 12" colour picture and the RCA expect a full colour 30" x 40" TV set by the end of the 1980's. The Sony Corporation lists among its design considerations for a true flat CRT that the technology developed should be applicable to colour whilst Sinclair are looking forward to a "three tube projection projection TV with a 50" diagonal full colour display of which it is said the optics and the electronics could fit into a shoe-box-sized unit projecting onto a wall mounted screen.

Conclusions

The notion of a flat CRT has been on the go since the early fifties, and such a device is still not available. It is clear that the manufacture of a commercially viable flat CRT poses many problems. This is not surprising since the development of a new complex consumer product involves many problems in production engineering. Such a product must have many advantages to compete with rivals or it must be unique such as the first calculator or the first TV. The flat CRT is not a new idea and it has to compete with well established TV tubes. Nevertheless, the use of electron beams as a picture imaging device has few serious competitors at present. Other techniques such as liquid crystal displays, and individually addressable solid state cells, do not appear serious contenders to the CRT. Bearing in mind that the electron beam may be used to image single atoms as in the transmission electron microscope, the flat CRT may yet see the light of day.

References

D. Gabor, P. R. Stuart, and P. G. Kalman "A New Cathode-Ray Tube for Monochrome and Colour Television", Proc. IEE, *105*, Pt B, No. 24, 1958, p.581.

W. R. Aiken, "A Thin Cathode-Ray Tube", Proc. IRE, Dec., 1957, p.1599.

A. Ohkoshi, H. Sato, T. Nakano, T. Natori and M. Hatanaka "A Compact Flat Cathode Ray Tube" IEEE Transactions on Consumer Electronics Vol. CE-28, No. 3, August 1982, p.431.

C. Sinclair. "Small Flat Cathode Ray Tube", Society for Information Display, Int'l Symposium Digest of Tech. Papers, Vol. 12, 1981, p.138.

D. Lamport, A. Woodhead and A. Knapp SID 82 Digest, 18.3, 210.

Budget Account

by Gordon Stanley

Introduction and operation

This program was designed to help you keep track of a typical "budget account" which is offered by most of the major banks. Initially the bank gives you a form detailing various annual commitments e.g. rates, electricity, gas etc. You then estimate your total annual expenditure under each of the headings, add it all up, and go, cap in hand, to your bank.

Your friendly manager then adds his seal of approval, you make out a standing order for an amount 1/12th of the annual top plus a service charge. For the purpose of this program we are not interested in a service charge.

The program can be adapted to suit many kinds of similar applications needing mainly alterations to print statements and sizes of arrays.

On running the program for the first time, you will see a precautionary notice warning that having RUN the program all data will have been lost. If you did not intend to start from scratch you will now have to reload the program from the tape, however, if you are starting afresh, or for the first time, then pressing ENTER will guide you to the initialisation routine.

The first prompt asks for total annual budget, remember this should not include service charge. It is worth stating at this time that the monthly credit calculated by this program will not be the same as the standing order you make out to the bank which will include the service charge.

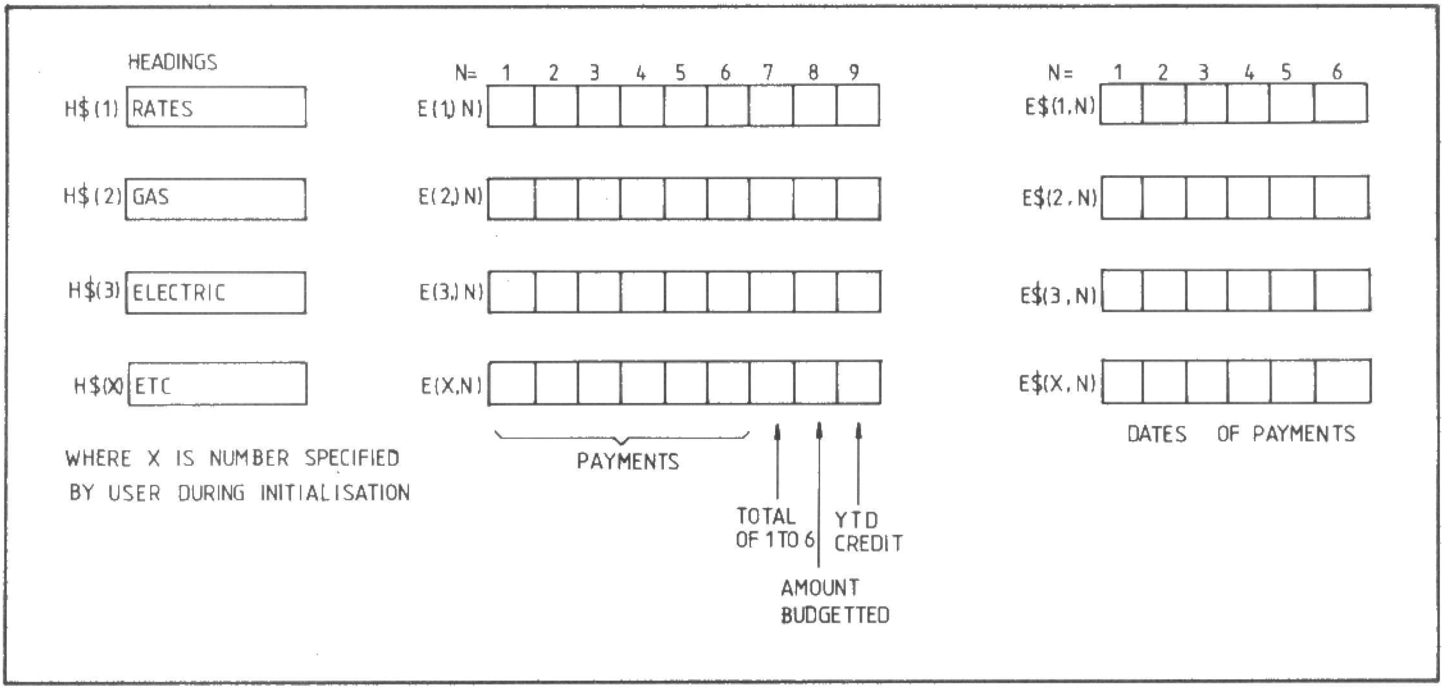
The next prompt asks for the number of headings. This, theoretically, is limited only by the storage size of your machine, but if you want to use all the facilities of this program, notably the bar graph summary by heading, then this will have to be limited to 13, or 16 if you don't mind the scale overprinting part of the graph. If more than 20 are used, the main menu routine will have to be rewritten.

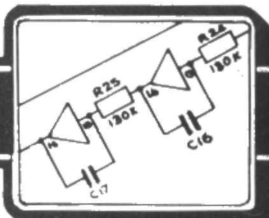
The next series of prompts ask for total annual budget, then the name and amount for each heading.

You will now go automatically to the date routine, more of which later. Briefly the program automatically calculates all credits from the date and it is essential that the date is set before making any payments from the account.

After setting the date, you will see the main menu. Numbers less than 20 will display payments, credits and balance of the selected heading, plus the overall account balance. You will also be able to make payments from the selected heading.

From now on it's all plain sailing. Use of the main menu options 21 and above are self explanatory, and use of these to save and verify is far safer than saving the program manually as the program will always restart at the correct point.





On small problem which complicated the writing of this program, and the calculation of credits from date is that not all time periods conveniently start on the 1st January. In my case the first month of the 12 month period was October, hence the little routine starting at line 2900. The date has been input in the format DD space MM space YY, where DD equals 2 digits for the day, MM month, and YY year. Hence if D$ was to equal "Ø1 Ø2 83" then the month can be found by looking at the value of B$ (4 TO 5), in this case 2. However as my year started in October then I am in the 5th month, not the 2nd. This conversion is done by slicing A$ as defined at line 2910. You can see that the second pair of characters in A$ is Ø5, so depending upon which month your "year" starts, then A$ will need to be changed. For those lacking the stimulus to work it out see fig. 1. A similar conversion is done at line 2940 to load the credits for each month into the appropriate array element used by the monthly bar graph summary (see fig. 2).

The working core of the program is the two dimensional array "E", where the number of first dimensions is set at initialisation time by the user. In the second dimension, the first six elements are payments, the seventh is the total of the first six, the eighth is the amount budgetted for this heading, and the ninth is the year to date credit.

Should you wish to alter the possible number of payments for each heading, this can be done by altering line 4060, also some statements in the main routine from line 1200 to 1970.

```
            BUDGET A/C

1   RATES            2   GAS
3   ELECTRIC         4   PHONE
5   CAR TAX          6   CAR INS
7   U.RATE

21   SET DATE
22   SAVE TO TAPE
23   VERIFY TAPE
24   SUMMARY BY HEADING
25   SUMMARY BY MONTH
26   STOP
27   INITIALISE
```

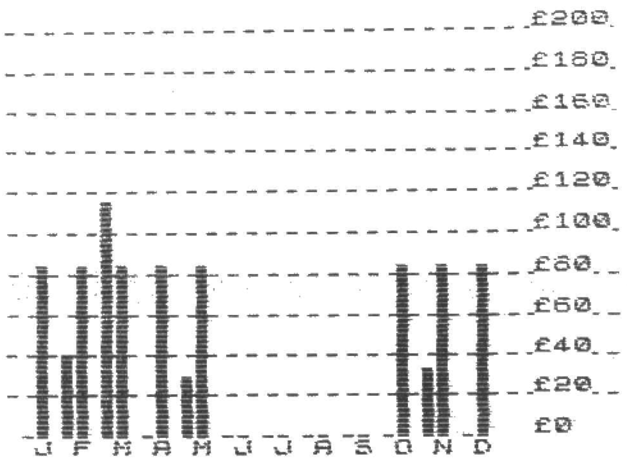
```
ELECTRIC    PAYMENTS          14 05 83
   DATE                         AMOUNT

   10 11 82                     £33.10
   12 02 83                     £40.13
   14 05 83                     £28.65

   TOTAL                       £101.88

AMOUNT BUDGETTED               £120.00

BALANCE                         £18.12

YTD CREDIT                      £80.00

YTD BALANCE                    £-21.88

OVERALL BALANCE                £450.20
```

FIGURE 1

FIRST MONTH	A$
JAN	010203040506070809101112
FEB	120102030405060708091011
MAR	111201020304050607080910
APR	101112010203040506070809
MAY	091011120102030405060708
JUN	080910111201020304050607
JUL	070809101112010203040506
AUG	060708091011120102030405
SEP	050607080910111201020304
OCT	040506070809101112010203
NOV	030405060708091011120102
DEC	020304050607080910111201

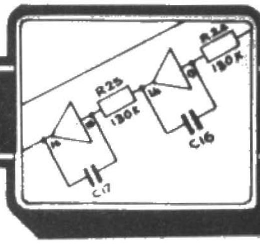
FIGURE 2

FIRST MONTH	H$
JAN	010203040506070809101112
FEB	020304050607080910111201
MAR	030405060708091011120102
APR	040506070809101112010203
MAY	050607080910111201020304
JUN	060708091011120102030405
JUL	070809101112010203040506
AUG	080910111201020304050607
SEP	091011120102030405060708
OCT	101112010203040506070809
NOV	111201020304050607080910
DEC	120102030405060708091011




```
   1 REM BUDGET A/C  © COPYRIGHT
   GORDON STANLEY NOVEMBER 1982
   5 POKE 23609,25
  10 LET S1=5100: LET R=100
  20 LET C=0
  30 LET D$=""
  40 INK 0: PAPER 7: BORDER 7
  50 LET MAX=C: DIM U(12): DIM H
(12)
  80 PRINT "IF YOU READ THIS AND
 ARE NOT     STARTING FROM SCRATC
H YOU HAVE  JUST LOST IT!!!!": I
NPUT "PRESS ENTER TO CONTINUE ";
A$
  90 GO TO 4000
 100 CLS : PRINT TAB 11;"BUDGET
A/C"''
 110 FOR I=1 TO H STEP 2
 120 PRINT I;TAB 3;H$(I);
 130 IF I+1>H THEN GO TO 150
 140 PRINT TAB 16;I+1;TAB 19;H$(
I+1)
 150 NEXT I
 160 PRINT ''
 200 PRINT 21;TAB 4;"SET DATE"
 210 PRINT 22;TAB 4;"SAVE TO TAP
E"
 220 PRINT 23;TAB 4;"VERIFY TAPE
"
 230 PRINT 24;TAB 4;"SUMMARY BY
HEADING"
 240 PRINT 25;TAB 4;"SUMMARY BY
MONTH"
 250 PRINT 26;TAB 4;"STOP"
 260 PRINT 27;TAB 4;"INITIALISE"
 390 INPUT PAPER 6; FLASH 1;"SEL
ECT OPTION";A
 400 IF A>20 AND A<28 THEN GO TO
 ((A-21)*200)+2800
 410 IF A<1 OR A>H THEN GO TO R
1200 CLS : PRINT PAPER 1; INK 7;
H$(A);" PAYMENTS";TAB 24;D$
1210 PRINT 'TAB 3;"DATE";TAB 26;
"AMOUNT"
1250 PRINT
1300 LET E(A,7)=0
1305 FOR J=1 TO 6
1310 IF E$(A,J)="          " THEN
LET J=6: GO TO 1360
1315 LET E(A,7)=E(A,7)+E(A,J)
1320 LET B$=STR$ E(A,J)
1330 GO SUB S1
1340 PRINT TAB 3;E$(A,J);TAB 31-
LEN B$;"£";B$
1360 NEXT J
1370 LET B$=STR$ E(A,7): GO SUB
S1
1380 PRINT 'TAB 3;"TOTAL";TAB 31
-LEN B$;"£";B$
1410 LET B$=STR$ E(A,8)
1420 GO SUB S1
1430 PRINT '"AMOUNT BUDGETTED";T
AB 31-LEN B$;"£";B$
1440 LET B$=STR$ (E(A,8)-E(A,7))
: GO SUB S1
1450 IF E(A,8)-E(A,7)<0 THEN PAP
ER 2: INK 7
1460 PRINT '"BALANCE";TAB 31-LEN
 B$;"£";B$
1470 PAPER 7: INK 0
1500 LET E(A,9)=E(A,8)/12*NM
1510 LET E(A,9)=INT (E(A,9)*100)
/100
1530 LET B$=STR$ E(A,9): GO SUB
S1
1540 PRINT '"YTD CREDIT";TAB 31-
LEN B$;"£";B$
1550 LET B$=STR$ (INT ((E(A,9)-E
(A,7))*100+.05)/100): GO SUB S1
1570 IF (E(A,9)-E(A,7))<0 THEN P
APER 2: INK 7
1580 PRINT '"YTD BALANCE";TAB 31
-LEN B$;"£";B$
1590 PAPER 7: INK 0
1600 LET C=0
1610 FOR I=1 TO H
1620 LET C=C+E(I,7)
1630 NEXT I
1650 LET C=(NM*MC)-C
1660 LET B$=STR$ C
1670 GO SUB S1
1680 IF C<0 THEN PAPER 2: INK 7
1690 PRINT '"OVERALL BALANCE";TA
B 31-LEN B$;"£";B$
1700 PAPER 7: INK 0
1710 INPUT PAPER 6; FLASH 1;"PAY
MENT=P COPY=C RETURN=R ";A$
1720 IF A$="R" THEN GO TO R
1730 IF A$="C" THEN COPY : GO TO
 R
1740 IF A$<>"P" THEN GO TO 1710
1800 FOR J=1 TO 6
1810 IF E$(A,J)="          " THEN
GO TO 1900
1820 NEXT J
1830 CLS : PRINT PAPER 2; INK 7;
AT 10,0;"NO ROOM FOR ANY MORE PA
YMENTS"
1840 PAUSE 200
1850 GO TO R
1900 CLS : PRINT PAPER 1; INK 7;
AT 4,10;H$(A)
1910 PRINT PAPER 6; FLASH 1;AT 1
0,10;"ENTER AMOUNT"
1920 INPUT E(A,J)
1930 LET B$=STR$ E(A,J): GO SUB
S1
1940 PRINT AT 12,12;"£";B$
1950 PRINT PAPER 6; FLASH 1;AT 1
4,6;"IS THIS CORRECT Y/N?"
1960 INPUT A$
1970 IF A$<>"Y" THEN LET E(A,J)=
0: GO TO 1200
2000 LET E$(A,J)=D$
2020 LET U(Y)=U(Y)+E(A,J)
2090 GO TO 1200
2790 STOP
2800 CLS : POKE 23609,25
2810 PRINT AT 4,4;"PREVIOUS DATE
 ";D$; PAPER 6; FLASH 1;AT 8,7;"
ENTER TODAYS DATE";AT 12,12;"DD
MM YY"
2830 INPUT B$
2840 PRINT AT 12,12;B$
2850 PRINT AT 16,4;"IS THIS CORR
ECT? Y OR N"
2860 INPUT A$
2870 IF A$<>"Y" THEN GO TO 2800
2880 LET A$=B$(4 TO 5)
```

```
2890 IF A$<>"01" AND A$<>"02" AND A$<>"03" AND A$<>"04" AND A$<>"05" AND A$<>"06" AND A$<>"07" AND A$<>"08" AND A$<>"09" AND A$<>"10" AND A$<>"11" AND A$<>"12" THEN PRINT INK 2; FLASH 1;AT 16,4;"     MONTH INCORRECT      ": PAUSE 100: GO TO 2800
2900 LET D$=B$: LET Y=VAL B$(4 TO 5)
2910 LET A$="040506070809101112010203"
2920 LET NM=VAL A$(Y*2-1 TO Y*2)
2930 LET F=NM*MC
2940 LET A$="101112010203040506070809"
2950 FOR I=1 TO NM
2960 LET W(VAL A$(I*2-1 TO I*2))=MC
2980 NEXT I
2990 GO TO R
3000 LET A$="": LET A=0: CLS
3010 SAVE "BUDGET" LINE 2800
3020 GO TO R
3200 LET A=0: CLS
3210 PRINT AT 10,2; PAPER 6; FLASH 1;"START TAPE THEN PRESS ENTER"
3220 INPUT A$
3230 VERIFY "BUDGET"
3240 GO TO R
3400 LET MAX=0: INK 6: CLS
3405 FOR I=1 TO H
3410 LET E(I,9)=INT (E(I,8)/12*NM*100)/100
3415 IF E(I,9)>MAX THEN LET MAX=E(I,9)
3420 IF E(I,7)>MAX THEN LET MAX=E(I,7)
3425 NEXT I
3430 FOR I=100 TO 10000 STEP 100
3435 IF I>MAX THEN LET MAX=I: LET I=9E9
3440 NEXT I
3445 LET PP=20: PAPER 0: BORDER 1: CLS
3450 FOR I=2 TO H*2 STEP 2
3455 LET V=4+(I-2)*5
3460 INK 6
3463 IF INT (I/4)*2=I/2 THEN INK 3
3465 PRINT ;AT 21,I-2;I/2: BRIGHT 1
3470 FOR J=V TO V+3
3475 INK 2: PLOT J,8
3480 DRAW 0,160*(E(I/2,7)/MAX)
3485 INK 4: PLOT J+6,8
3490 DRAW 0,160*(E(I/2,9)/MAX)
3495 NEXT J
3500 NEXT I
3502 BRIGHT 0
3505 FOR I=24 TO 168 STEP 16
3510 FOR J=7 TO 250 STEP 8
3515 INK 8
3520 PLOT J-1,I: DRAW 3,0
3525 NEXT J
3530 NEXT I
3535 FOR I=0 TO MAX+1 STEP MAX/10
3540 PRINT INK 5;AT PP,27;"£";I
3545 LET PP=PP-2
3550 NEXT I
3590 GO TO 3770
3600 LET MAX=0: INK 6: CLS
3605 FOR I=1 TO 12
3610 IF U(I)>MAX THEN LET MAX=U(I)
3615 IF W(I)>MAX THEN LET MAX=W(I)
3620 NEXT I
3625 FOR I=100 TO 10000 STEP 100
3630 IF I>MAX THEN LET MAX=I: LET I=9E9
3635 NEXT I
3640 RESTORE 3645: LET PP=20: PAPER 0: BORDER 1: CLS
3645 DATA "J","F","M","A","M","J","J","A","S","O","N","D"
3650 FOR I=2 TO 24 STEP 2
3655 LET V=4+(I-1)*8
3660 READ A$
3665 PRINT INK 6;AT 21,I;A$: BRIGHT 1
3670 FOR J=V TO V+3
3675 INK 2: PLOT J,8
3680 DRAW 0,160*(U(I/2)/MAX)
3685 INK 4: PLOT J+6,8
3690 DRAW 0,160*(W(I/2)/MAX)
3695 NEXT J
3700 NEXT I
3702 BRIGHT 0
3705 FOR I=24 TO 168 STEP 16
3710 FOR J=7 TO 250 STEP 8
3715 INK 8
3720 PLOT J-1,I: DRAW 3,0
3725 NEXT J
3730 NEXT I
3735 FOR I=0 TO MAX+1 STEP MAX/10
3740 PRINT INK 5;AT PP,27;"£";I
3745 LET PP=PP-2
3750 NEXT I
3770 INPUT "RETURN=R COPY=C ";A$
3775 INK 0: PAPER 7: BORDER 7
3780 IF A$="C" THEN COPY : GO TO R
3785 IF A$<>"R" THEN GO TO 3770
3799 GO TO R
3800 STOP
4000 CLS : LET A=0
4010 PRINT AT 2,8;"INITIALISATION"
4020 PRINT AT 6,3; PAPER 6; FLASH 1;"ENTER TOTAL ANNUAL BUDGET"
4025 PRINT AT 10,5; PAPER 6; FLASH 1;"OR ENTER 'R' TO RETURN"
4030 INPUT A$
4035 IF A$="R" THEN GO TO R
4040 LET TB=VAL A$
4045 LET MC=INT ((TB/12)*100)/100
4050 CLS : PRINT AT 10,5; PAPER 5; FLASH 1;"HOW MANY HEADINGS?"
4055 INPUT H
4060 DIM H$(H,10): DIM E$(H,6,5): DIM E(H,9)
4065 CLS : PRINT PAPER 5;TAB 5;"ANNUAL BUDGET = £";TB,
4070 FOR I=1 TO H
4075 PRINT '"HEADING ";I;" ?"
4080 INPUT H$(I)
4085 PRINT "AMOUNT FOR ";H$(I);" ? ";
4090 INPUT E(I,8)
4095 LET A=A+E(I,8)
4100 PRINT E(I,8)
4105 NEXT I
4110 IF A<>TB THEN CLS : PRINT AT 10,0; PAPER 2; INK 7;" HEADINGS DO NOT EQUAL TOTAL ": PAUSE 200: GO TO 4000
4130 PRINT ' PAPER 6; FLASH 1;"IS ALL CORRECT? Y OR N"
4140 INPUT A$
4150 IF A$="Y" THEN GO TO 2800
4160 IF A$<>"N" THEN GO TO 4140
4170 GO TO 4000
5100 FOR I=1 TO LEN B$
5110 IF CODE B$(I)=46 AND I+2=LEN B$ THEN GO TO 5190
5120 IF CODE B$(I)=46 AND I+1=LEN B$ THEN GO TO 5160
5130 NEXT I
5140 LET B$=B$+".0"
5160 LET B$=B$+"0"
5190 RETURN
```



Understanding Digital Electronics

by J. Oliver Linton

ARTICLE 2 — Combinational Logic

The easiest and most enjoyable way to get to grips with digital electronics is to experiment with real components and to design simple circuits to solve simple but realistic problems. The following basic kit of parts will enable you to explore the ideas presented in this and subsequent articles (though only those integrated circuits relevant to this month's article are listed here).

1	Breadboard	(eg CSC EXPERIMENTOR 300)
1	6v battery	(eg PJ996 or 4 x SP2)
1	Pair battery clips	
1	1N4001 diode	
4	Red LED's	
4	BC108 transistors	
4	33k resistors	
3	7400	(quad 2 input NAND)
1	7402	(quad 2 input NOR)
1	7404	(hex inverter)
1	7447	(7 segment decoder)
1	7 segment LED display	(common **anode**)
7	220 ohm resistors	

Since TTL integrated circuits have a maximum input voltage rating of 5.5 V, a fresh 6 V battery could possibly harm them. To avoid this, place a diode in series with a battery as shown in fig. (1). This will drop the voltage by about 0.6 V and also provide protection against connecting the battery the wrong way round.

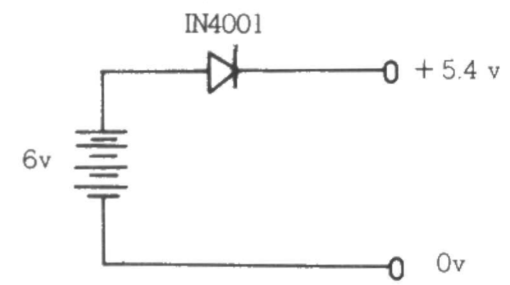


Fig. 1 Power Supply

It is also convenient to have ready built on your breadboard a number of display units consisting of the circuit shown in fig. (2). A TTL gate will happily sink 10 mA or so when in the LOW state, but it is only capable of sourcing a few mA when HIGH. An LED needs about 10 mA to make it glow brightly – hence the need for the transistor. The LED will glow when the input X is HIGH, and by connecting this input to the appropriate point you can examine the output of any gate at will.

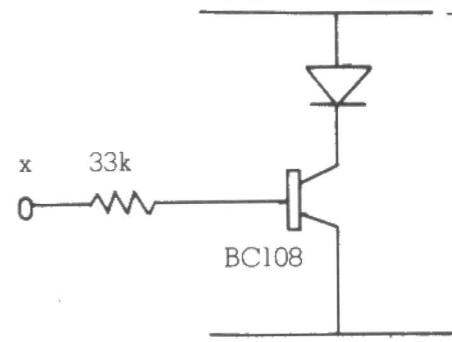


Fig. 2 LED indicator

Now, before we get back to the real business, the solution to the problem of converting a 7400 (quad 2 input NAND) into a single 2 input NOR gate is shown in fig. (3). The inverters can of course be made out of NAND gates simply by joining the two inputs of the NAND gate together.

We ended last month with the design of a 2 channel multiplexer. The 7400 series of logic gates provides us with a number of more complex multiplexers from which to choose.

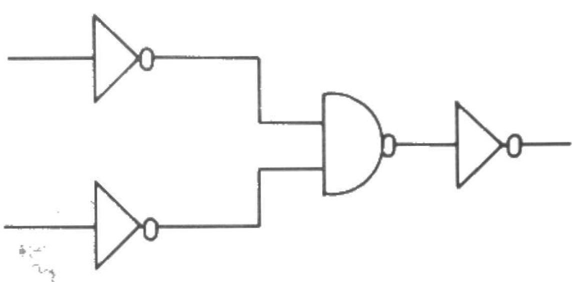
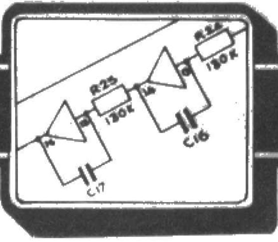


Fig. 3 2 input NOR gate

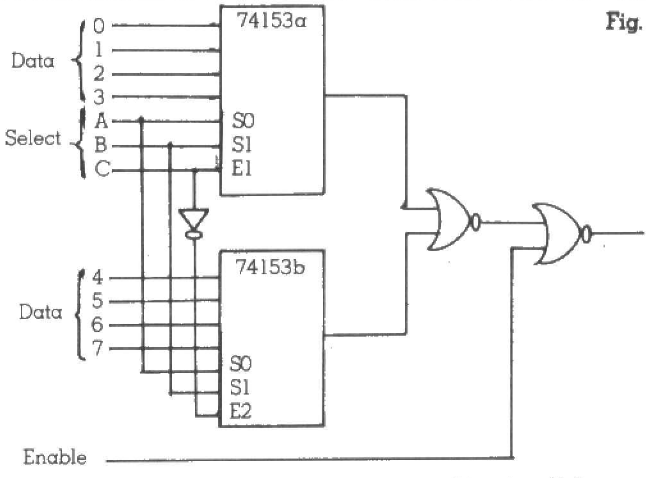


74157 Quad 2 input (1 bit) multiplexer
74153 Dual 4 input (2 bit) multiplexer
74151 8 input (3 bit) multiplexer
74150 16 input (4 bit) multiplexer

In effect, multiplexers are multiway switches controlled by a binary address placed on the select lines (often labelled S0, S1, S2 etc.). One address line can control 2 inputs as we saw last month. 2 address lines can control 4 inputs and 3 address lines, 8 inputs etc. In addition there is always an enable line (usually labelled $\overline{E}$) which enables the output when held LOW. This pin permits several chips to be linked together as in the example in fig. (4) where a dual 4 input multiplexer has been turned into an 8 input multiplexer. Note that the most significant address line (C) has been connected to the enable lines of both halves, but in one case through an inverter so that only one half is enabled at any one time. In the 74153 the select lines of the two halves are already connected internally. Since the outputs are LOW when disabled, they must be connected together by an OR gate. It is most convenient to use a NOR gate followed by a second NOR gate into which the $\overline{E}$ line of the new 8 channel multiplexer may be fed. In a similar way any number of small units may be chained together to make larger ones. This principle assumes great importance when designing memory circuits and we shall return to this in a later article. For the moment we shall observe that, instead of chaining multiplexers together, they can also be placed in parallel like a multi-pole switch. Two 74157's are often used in this way to switch a whole 8 bit data bus from one source to another like an 8 pole 2 way switch. As an exercise, try making a 4 input (2 bit) multiplexer out of NAND gates etc. I think you will need 9 NAND gates and a couple of inverters. (Hint – try coupling two 2 input multiplexers together).

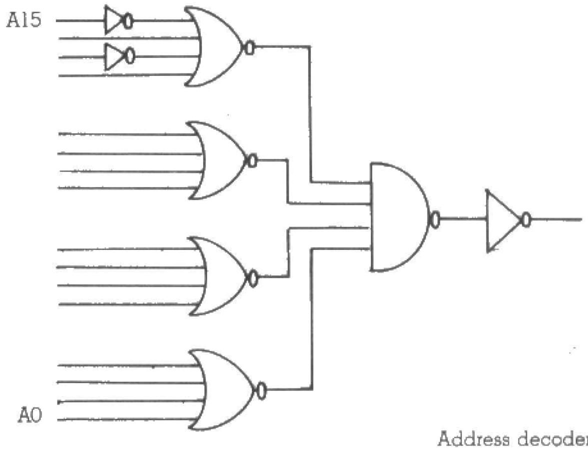
One of the important applications of combinational logic in a computer is in address decoding. A typical microprocessor may have 16 address lines A0 – A15. When the computer wants to access a particular memory location it places the number of the location, in binary of course, on these address lines with the least significant bit (LSB) on A0 and the most significant bit (MSB) on A15. Suppose the address to be located is number 40960. This forbidding looking number is rather simple in binary; it is 1010 0000 0000 0000 or, for those of you with sixteen fingers, A000 in hex. The address lines A15 and A13 would therefore be taken HIGH and all the rest LOW. The question is – how does the memory location know when it is being referred to? Somehow this unique combination of 0's and 1's must be decoded so that a unique point in one of the computer's memory circuits is activated. In practice, many memory locations share a good deal of decoding logic and much of it is provided on-chip; but if location 40960 was say an input port or a sound generator etc., then the circuit shown in fig. (5) would provide complete decoding. The inverters in lines 13 and 15 convert the 1's to 0's; the following 4 input NOR gates and the 4 input NAND gate detect the condition that all the inputs to the NOR gates are LOW; the final inverter is necessary to give a HIGH output when the correct address is decoded. Any address may be decoded in this way by putting inverters in the appropriate address lines. A number of TTL integrated circuits are available which provide full decoding for a small number of address lines, e.g.

Fig. 4



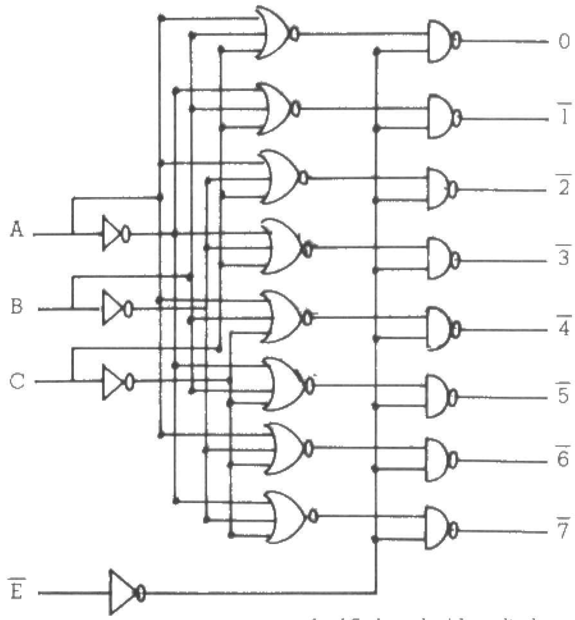
8 input multiplexer

Fig. 5



Address decoder

Fig. 6



1 of 8 decoder/demultiplexer

74139 Dual 1 of 4 decoder/demultiplexer
74138 1 of 8 decoder/demultiplexer
74154 1 of 16 decoder/demultiplexer

In the data sheets, the address lines are usually labelled A0, A1 etc. and the decoded outputs $\overline{0}$, $\overline{1}$, $\overline{2}$ etc. The bar over these numbers indicates that these outputs are normally HIGH and that they go LOW when the appropriate address is placed on the address lines.

Now, if you want something else to do with your breadboard and chips, design and build a 1 of 4 decoder – i.e. a logic circuit that will light up one of four LED's depending on the state of two address lines A0 and A1. You should be able to do this with a 7402 plus a couple of inverters from a 7404.

Another look at the pin diagrams of the above components will reveal some additional inputs labelled $\overline{E}$, $\overline{E1}$ or $\overline{E2}$ etc. Moreover, you will have noticed that they are also called demultiplexers as well as decoders. You have guessed it – these are enable lines which have to be set HIGh or LOW (depending on whether there is a bar over the $\overline{E}$ or not) for the output lines to be activated. But there is another way to look at the function of these lines. Consider the $\overline{E}$ line of a 1 of 4 decoder (74139). Suppose the number 01 is placed on the two address lines A1 and A0. When $\overline{E}$ is HIGH, output line $\overline{1}$ is HIGH like all the others. But when the $\overline{E}$ line is taken LOW the output line $\overline{1}$ also goes LOW. In other words $\overline{1}$ follows $\overline{E}$. If a stream of data is input onto the enable line, it will appear on whichever output is currently addressed by the address lines. This process whereby a data steam can be switched to one of several lines is called demultiplexing and is the reverse of the multiplexing process considered earlier. Again I urge you to design and build a 1 of 4 demultiplexer with enable line. This can be done by adding a 7400 to the 1 of 4 decoder (though the output will be inverted). A possible design for a complete 1 of 8 decoder/demultiplexer is shown in fig. (6).

There are a number of more specialized decoders in the 7400 series and some of these are listed below:

7447 BCD to 7 segment decoder (common anode)
7448 BCD to 7 segment decoder (common cathode)
74155 Dual 1 of 4 decoder/demultiplexer
7442 BCD to 1 of 10 decoder
74141 BCD to 1 of 10 decoder/driver

The 74155 differs from the 74139 only in having common address lines and two more enable lines. The 7447 is a particularly useful chip because it can drive a 7 segment display directly. In order to limit the current flowing through the LED's, however, it is necessary to put 220 ohm resistors between the I.C. and the display as shown in fig. (7). The outputs of the 7447 are active LOW and therefore need a display whose LED's all have a common anode (i.e. positive terminal).

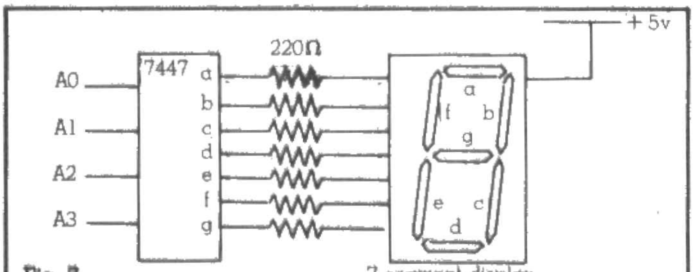


Fig. 7

The 7447 has three other interesting pins. $\overline{LT}$, $\overline{RBI}$ and $\overline{RBO}$. $\overline{LT}$ stands for Lamp Test and when taken LOW, lights all the LED's regardless of the input. $\overline{RBI}$ and $\overline{RBO}$ stand for Ripple Blanking In and Ripple Blanking Out and are used to suppress leading zero's in a multi-digit display. When $\overline{RBI}$ is LOW, the display will not show the number zero. Moreover, when the display is thus blanked out, $\overline{RBO}$ goes LOW. If $\overline{RBO}$ of one display driver is connected to $\overline{RBI}$ of the display driver of the next least significant digit, then all leading zero's will be suppressed. To start the ripple going, $\overline{RBI}$ of the most significant digit must be connected to ground.

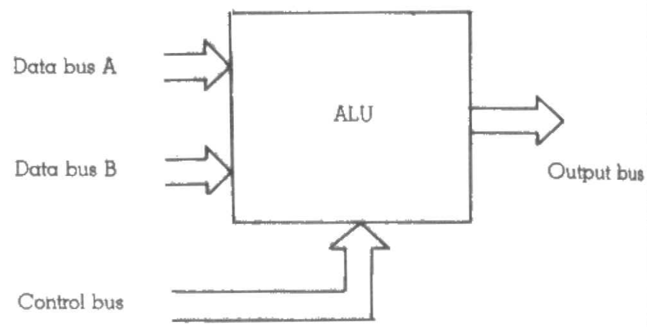


Fig. 8

Arithmetic Logic Unit

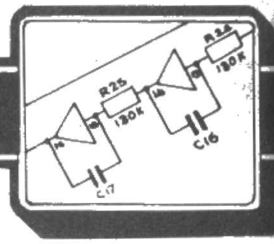
The opposite of a decoder is an encoder – i.e. a device which outputs a binary number according to which of a number of input lines is selected. Clearly such a device must cope with the possibility that more than one input line may be selected simultaneously. In this case the highest number is given priority – hence the name 'priority encoder'. The 74147 is a useful device which encodes 10 lines into 4. It could be used in a computer controlled robot with a number of sensors. The most urgent sensors can be given higher priority and will be dealt with first. Try making a 3 to 2 line encoder. it can be done with 3 NAND gates and 3 inverters.

I think most people would agree that the most important application of combinational logic in a computer is in the Arithmetic Logic Unit (ALU) where all the number crunching is done. A simple block diagram of such a unit is shown in fig. (8). In a small microprocessor, the input and output buses will be 8 bits wide (plus a few extras like carry lines etc.). The number of control lines needed depends on the number of different operations which the ALU can carry out. Most ALU's will at least AND, OR, ADD, SUBTRACT and COMPLEMENT the inputs and on larger machines even multiplication and division can be done by the hardware.

How is it that one unit can do so many different things? Let us try to make the principles clearer by considering the design of a unit that will take two 1-bit inputs and perform either an AND, OR, NAND or NOR function on them depending on the state of two control lines as follows:

C1	C0	Operation
0	0	AND
0	1	OR
1	0	NAND
1	1	NOR

Continued on Page 53



THE BBC MICRO AND THE LIGHT FANTASTIC:

OPTICAL INTERFERENCE PATTERNS DISPLAYED USING THE BBC MICRO

by Dr. A. R. Allen, Norwich City College

Microcomputers are being used increasingly in the physics lab. in schools and colleges. With minimal extra circuitry it is possible to use the micro to monitor directly the values of physical variables. Light intensity is an example of a useful quantity to get the computer to record. I describe below a simple arrangement whereby an interference or diffraction pattern may be scanned and the intensity distribution graph displayed. This provides a very effective demonstration which would be difficult to do without a microcomputer. It may also be used as the basis of a student experiment in advanced courses.

The set-up is shown in Figure 1. The light source is a laser: low power He-Ne lasers are becoming more usual pieces of equipment in many schools. A slit produces a diffraction pattern, the intensity of which is sampled using a photodiode. After suitable amplification, the voltage is digitized and read by the computer.

The light detector must be scanned across the interference pattern. In order that the computer may relate light intensity to the position of the sensor, the latter should ideally be under computer control. This could be achieved using a stepper motor arrangement. A chart recorder provides a convenient alternative: if the photodiode is mounted on the chart recorder pen holder, this may be moved along by applying a steadily increasing voltage to the recorder. The computer may do this if a digital-to-analog converter is connected to the user port.

The current through a reverse-biassed photodiode is directly proportional to the intensity of the light falling on it, and if it is connected to the input of a FET op. amp. such as the LF355 (see Figure 2), the output voltage is in proportion to the light intensity.(*) The output of the op. amp. is $V_o = I_D R_F$, where I_D is the photodiode current and R_F is the feedback resistor. The values of R_F shown give a useful range of operation for output voltages around 1 V.

This voltage is read by channel 1 of the analog-to-digital converter in the BBC Micro. This is designed to convert analog signals in the range 0 to about +1.8 V, and the op amp gain should be adjusted accordingly. It is wise to protect each analog input with a zener diode, as shown in Figure 3. (If at any time you are unlucky enough to input a large voltage and the μPD7002 a-to-d converter appears to die, then try switching the BBC Micro off and back on again before sending off for a replacement chip. This worked for me when a charged capacitor was inadvertently connected before the analog input were protected!)

The user port (PB) is incremented with a suitable delay after each value, driving the chart recorder via a d-to-a converter. a standard converter is used in this design, see Figure 4. This gives an output voltage in the range 0 to +3.83 V using the internal voltage reference of the ZN425 and the op amp as shown.

The user may offset the baseline of the intensity graph, by entering the number of vertical axis divisions by which the baseline is to be offset (default is zero offset). This is a useful feature for compensating for ambient light levels. It also allows one to superimpose patterns without confusion. After the graph is drawn, the screen may be cleared or the next pattern may be superimposed.

If a permanent record of the pattern is required but you do not have access to a printer, then the chart recorder may be used again – this time in its normal function! When the 'chart rec.' option is selected, the data is outputted through the user port to the d-to-a converter at a rate suitable for a pen recorder to follow.

The intensity graph being gradually revealed as the sensor traverses the pattern can be an effective demonstration. If the chart recorder is arranged correctly, the diffraction pattern may be seen at the same time as the graph is being displayed. Several features of the single slit diffraction pattern may be observed (Figure 5). For example, the central maximum is twice the width of the subsidiary maxima. The graphs also demonstrate how rapidly the height of these peaks decreases: 1.0, 0.047, 0.017, 0.008 starting from the centre. Since there is such a wide range of intensity in some patterns, for certain purposes it may be preferable for the graph to be a logarithmic one. This could be achieved using a logarithmic amplifier in place of the photodiode op amp, but is done in software using the INTERF program given here. While distorting the shape of the graph, it does make visible some otherwise insignificant maxima. A log response is included as an option (default is linear). This figure also shows the effect of changing the slit width: a narrower slit resulting in a broader pattern. Figure 6 shows 'Young's slits' 1983 style! By superimposing, it is possible to show that the envelope of the double slit interference pattern is the diffraction pattern due to a single slit of the same width as one of the slits (Figure 7). A screen dump to a printer can be photocopied for students to analyse.

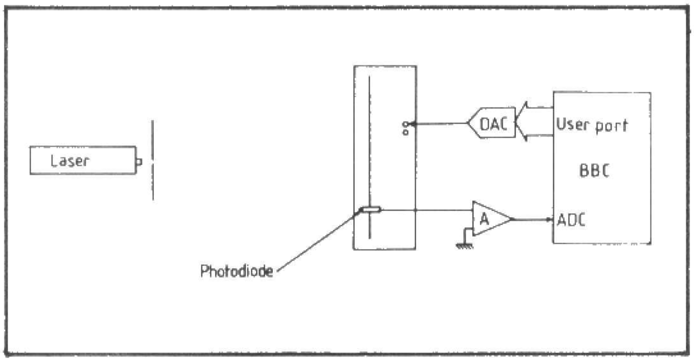


Figure 1
Arrangement for scanning optical diffraction patterns under computer control.

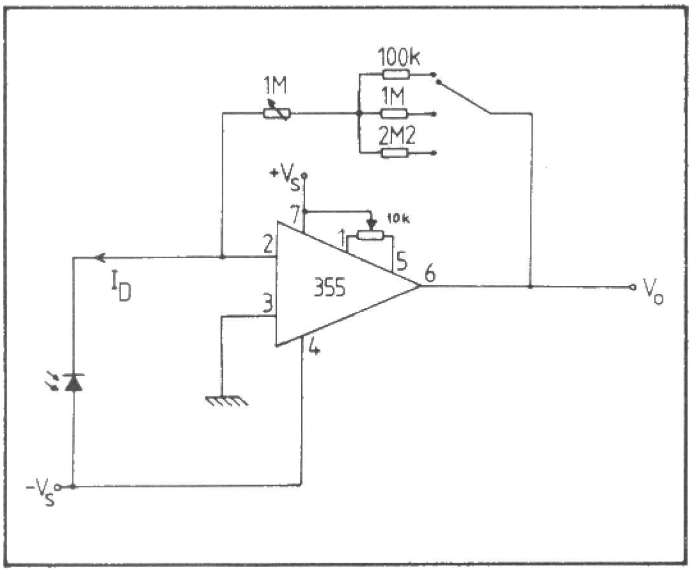


Figure 2
Linear photometer circuit. $V_s = \pm 4$ to ± 18 V.

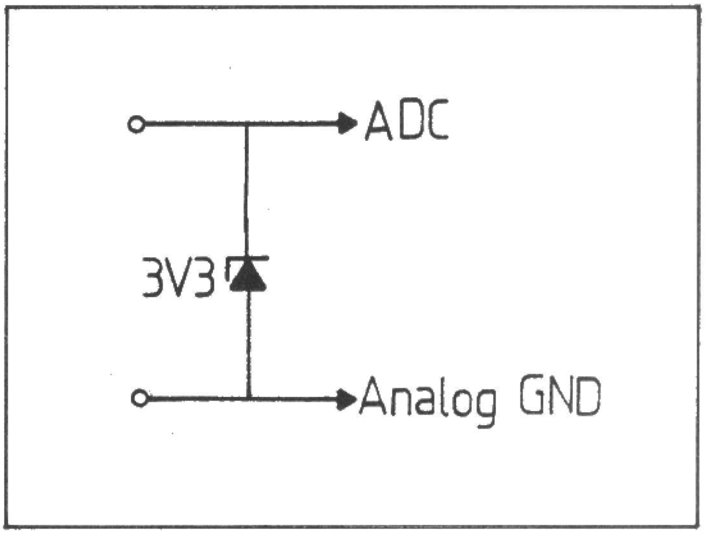


Figure 3
Analog input protection

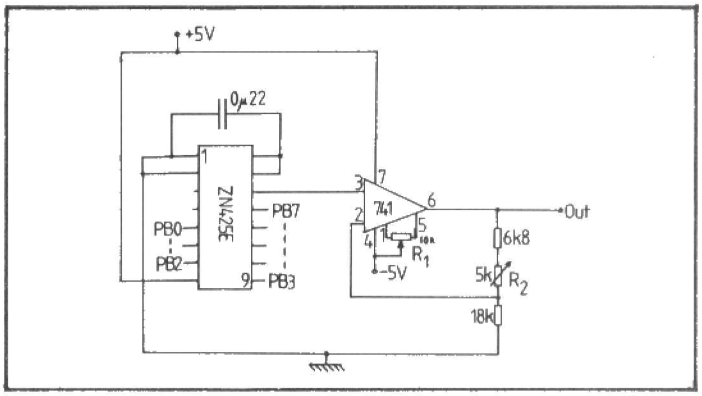


Figure 4
Digital-to-analog converter. R_1 — set zero; R_2 — set full-scale

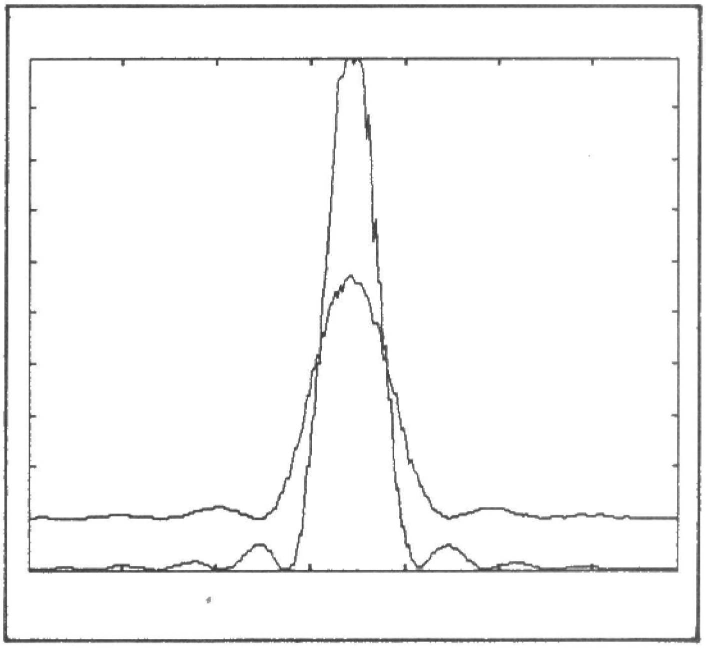


Figure 5
Diffraction patterns for slits of two widths

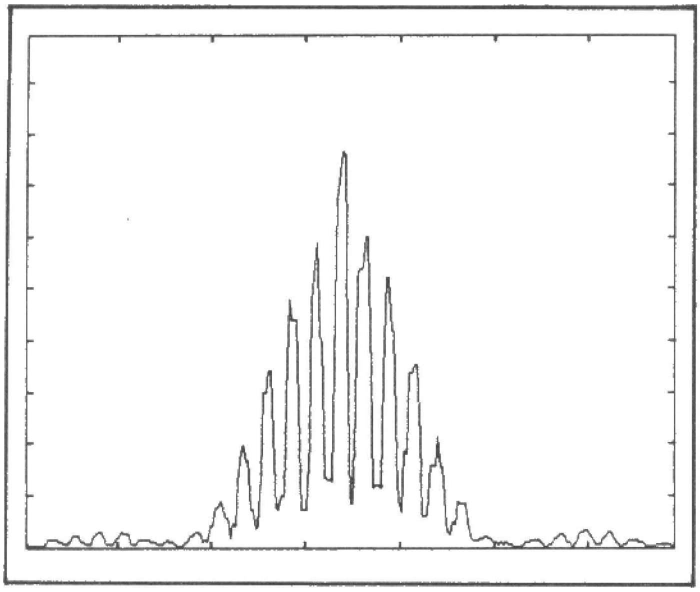


Figure 6
Double slit interference pattern

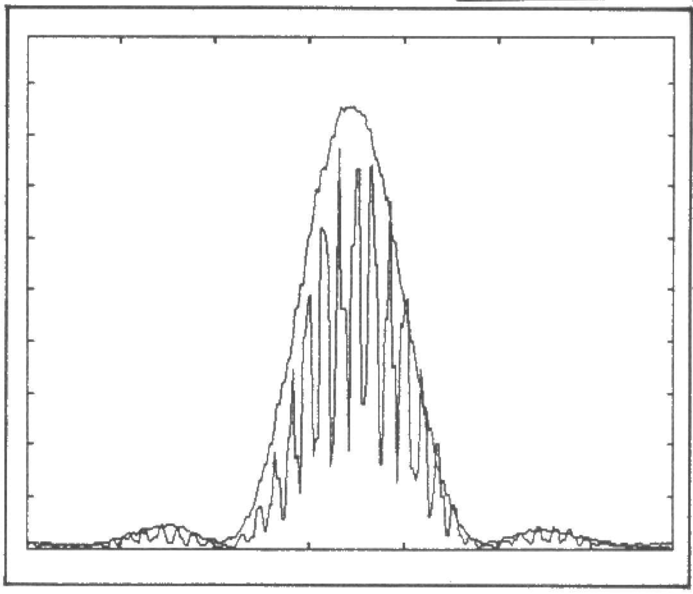


Figure 7
Single and double slit patterns superimposed

BBC Micro as a Storage Scope

Thank you to all those who have written in expressing interest in this project and have said you are building it. There have been one or two recurring queries which I shall answer now.

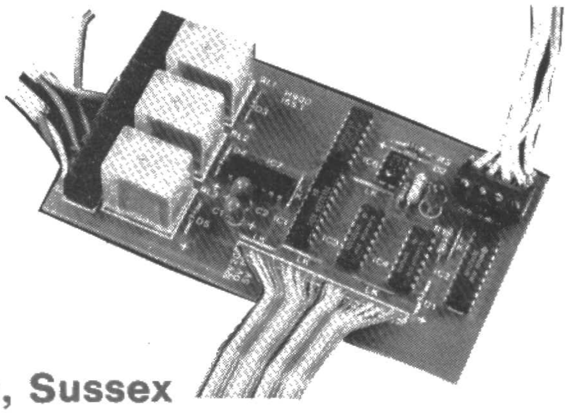
In Figure 2 of the Cecember '82 article, the oscillator providing the clock for the converter is a 74LS124. This chip is readily available from electronics components distributors (e.g. see Watford Electronics or Technomatic adverts in this magazine). In order to obtain an output frequency of 1MHz, the value of C_{ext} should be 100 pF. On the analog-to-digital converter ZN427E, pins 1 and 2 should be connected together (as shown) and there is no external connection to EOC. In fact, EOC is low during a conversion and goes high to signal end of conversion. So you could use this to send a signal to an edge-sensitive control line of the VIA that the data is ready: this feature was not made use of in this design. Lastly, the storage scope program as published works on both Operating Systems 0.1 and 1.0.

(*) The diode used is the RS high speed photodiode (for details see RS data sheet 2135).

Program 1
Program for scanning optical interference patterns

```
 10 REM***INTERF***
 20 REM A R Allen.  Nov. 82
 30 REM Displays optical interference/
    diffraction patterns
 40 REM Response may be linear or log
 50 REM With options of superimposing
    graphs and output to chart recorder
 60 DIMintensity(255)
 70 ?&FE62=255
 80 MODE0
 90 PROCscreen
100 REPEAT
110   ?&FE60=0
120   INPUT"Log response",log$
130   INPUT"Offset",offset
140   NOW=TIME
150   REPEAT UNTIL TIME>NOW+200
160   MOVE0,0
170   FOR position=0 TO 255
180     ?&FE60=position
190     intensity(position)=ADVAL(1)/64
200     IF log$="Y" THEN PROClog ELSE
        PROClin
210     NEXT position
220   PROCoptions
230   UNTIL 0
240
250 DEF PROCscreen
260 CLG:CLS
270 VDU28,1,8,16,1
280 DRAW0,1023
290 DRAW1279,1023
300 DRAW1279,0
310 DRAW0,0
320 VDU5
330 FOR X=0 TO 1264 STEP 1264
340   FOR Y=118.4 TO 1023 STEP 102.4
350     MOVEX,Y
360     PRINT"-"
370     NEXT
380   NEXT
390 FOR Y=15 TO 1035 STEP 1020
400   FOR X=176 TO 1280 STEP 185.5
410     MOVEX,Y
420     PRINT":"
430     NEXT
440   NEXT
450 VDU4
460 ENDPROC
470
480 DEF PROClog
490 IF intensity(position)<=0 THEN DRAW
    position*1280/255,offset*102 ELSE
    DRAW position*1280/2
55,LOG(intensity(position))*1024/LOG(1024)
    +offset*102
500 WAIT=TIME
510 REPEAT UNTIL TIME>WAIT+3
520 ENDPROC
530
540 DEF PROClin
550 DRAW position*1280/255,intensity
    (position)+offset*102
560 WAIT=TIME
570 REPEAT UNTIL TIME>WAIT+5
580 ENDPROC
590
600 DEF PROCoptions
610 PRINT"Clear screen? 1"
620 PRINT"Superimpose?  2"
630 PRINT"Chart rec.?   3"
640 INPUT R
650 IF R=1 PROCscreen
660 IF R=3 PROCchart:PROCoptions
670 ENDPROC
680
690 DEF PROCchart
700 REPEAT
710   INPUT"When ready,"'"enter Y " Y$
720   UNTIL Y$="Y"
730 PRINT"Outputting"
740 FOR V=0 TO 255
750   IF log$="Y" THEN ?&FE60=LOG
      (intensity(V))*256/LOG(1024)
      ELSE ?&FE60=intensity(V)/4
760   PAUSE=TIME
770   REPEAT UNTIL TIME>PAUSE+35
780   NEXT V
790 VDU7
800 ENDPROC
```

ZX Energy Management System



A. D. Chamier, B.Sc.,
Project Manager, Holec Energy, Horsham, Sussex

The increasing cost of energy throughout the 1970's was accompanied by a decreasing cost of electronic hardware. The ZX81 is a classic example. It became evident that the simple types of building heating controls installed up until the mid 1970's could be improved using the techniques previously only used by the aviation and defence industries, because of their high cost. Thus was born the concept of computer based Energy Management Systems.

A typical Energy Management System, such as supplied by the authors company, utilises a network of microprocessor based 'Outstations' interconnected by a simple two wire data highway, which may include British Telecom phone lines. The Outstations may be up to 1km apart. An 'Area controller', another microprocessor, issues instructions to the Outstations. Each individual Outstation is responsible for measuring temperatures and setting relay outputs to control the nearby heating systems as appropriate. A 'Supervisor' provides the user with 'Eyes' into the system, it communicates with the Area controllers, and the user is able to issue new instructions to the system, and receive back status reports, temperature graphs, historical trends, and Alarms from Plant malfunction. Central to all this is the ability to set temperatures of buildings individually to a unique predetermined plan for up to a year ahead. The Supervisor is a standard up-market Scientific desktop computer and may be disconnected from the system at any time without loss of control, and is then available to the user for other tasks.

A system as described above might consist of Supervisor, Area controller, and a dozen Outstations. Fitted in a medium size Hospital or factory the normal experience is that the fuel savings over the first two years equals or is more than the system cost. In other words an interest rate of 50%, which is good news for Industry and the NHS.

The author decided to look for a simple way of implementing the benefits of these techniques in the home. This was heavily stimulated by the family ZX81 being superseded by a ZX Spectrum!

The typical modern house has a Gas or Oil boiler feeding a domestic hot water tank (DHW) and radiators for room heating. As will be seen later the system described here is not limited to this kind of installation, enough flexibility is allowed in the system to permit most other heating systems to be connected.

At the most the existing controls would consist of: Time clock, Room 'stat, DHW tank 'stat, and the inevitable boiler water jacket 'stat. With these controls it is possible to wire a true 'Demand system' whereby when the Room and DHW are satisfied the Boiler and pumps etc. are Off until there is a new demand. But this level of control is seldom fitted, despite its low cost.

Even with the system described above there are various shortcomings. It does not permit weekday/weekend changes in lifestyle without continuous reprogramming the clock. It does not anticipate cold mornings, nor allow for mild ones. It does not allow for increased temperatures for morning getting up/bathtime and evening TV, with more modest temperatures during the day when an economic setting is acceptable. The same applies to the DHW where Bath/Laundry temperatures do not need to be sustained throughout the day.

From these thoughts this project was born.

Hardware

The three measurements of Room, Water and External air temperature are made with thermistors. The thermistors chosen represent a compromise on availability, price, accuracy. They are quite adequate and having no moving parts are very reliable.

The relay outputs are suitable for 240v mains switching of most domestic equipment. The highly inductive nature of boiler solenoids and circulating water pumps means that electrical interference suppression, using the recommended contact suppressors is mandatory, incidentally they dramaticallly reduce the wear on the relay contacts. The exact implementation may need some local experimentation, but fig. 1 shows a recommended scheme. The suppression ,may be installed at the relay terminals, rather than at the loads, with advantage. Use of capacitors without the damping resistor is not recommended, and ensure the capacitors are rated for 240v AC (240v DC is not the same) continuously. The effect of poor suppression is occasional lockout of the ZX81. No damage has been done to a ZX81 in the authors experience.

An installation retaining the existing control system is shown in fig. 2. Full reversion to the original system can be made in seconds, an important requirement if domestic harmony is to be retained during the initial phases! Operating the existing 'Stats to gross settings, and the clock off disables their action. In the authors installation the ZX81 and

controller reside on the airing cupboard floor with no ill effects. Monitoring is done from the Living room armchair via the family TV, courtesy of an aerial splitter in the loft!

The relay outputs are coded as Room demand, Water demand, and Room OR Water demand (logical OR). This can service most schemes, although the program is easily modified to individual needs.

The circuit consists of a simple 8 bit input/output port which responds to any address between 8192 and 16383. The ROM is masked from this space in usual fashion by D1. The entire address decoding is done by IC3 and IC4, including the Read/Write steering. Sinclair SPECTRUM owners may be pleased to hear that they are catered for. Delete IC4 (74LS00), also D1 and R1. Take the wire marked "not used" connection no. 5 to the $\overline{IORQ}$ on pin 15A of the Spectrum. Delete the connections marked $\overline{MREQ}$ and A15, and instead take both connections 6 and 7 to A5 on pin 21B of the Spectrum. The port is addressed from the Spectrum commands OUT and IN at address 65503.

Only one input line is used, D7, and the test of this bit being set is if the read value (PEEK) is greater than 127.

Output lines D0 and D1 are used to call each Thermistor into life by grounding its earthy end using the open collector outputs in IC5 (7403, not LS series as the Schottky clamps give a high bottoming voltage, use LS as a last resort). The two bits decode the three needed states. Output lines D4, D5, and D6 drive the relays via the high sink current rating IC7 (7438) which is not LS series here because of the current needed. The relays are nominally 12 volt but the standard Sinclair Power supply yields about 11 volts on load and is sufficient for the relays. Lower voltage relays need much higher coil currents.

The Analog to Digital converter is the Texas Instruments TL 507 which is delightfully simple to interface. With a resolution of 7 bits (128) the circuit is sequentially clocked with a negative pulse on pin 2. An internal DAC is compared with the analog input being measured, and the output on pin 4 goes to logic 0 to stop the clock. The number of clocks is the digital value of the analog input. The internal DAC is reset by taking pin 8 momentarily to logic 1. The whole operation is software controlled. R5 is necessary to ensure the 1 state crosses the generous Schmitt window in the TL507 clock input. The output is open collector and I used the pull up resistor to fit a LED so that some visual indicator of activity is available.

The ADC scaling is a function of the supply voltage between pins 6 and 3, nominally 5 volts. Fortunately the scaling is as a percentage, so the absolute voltage or its drift does not matter.

The upper value is 75% and the lower value is 25%.

The thermistors form the lower leg of a resistor chain through R2 from the 5 volt supply on pin 6, so all voltage variations are cancelled. For the value of R2 chosen (1K2) the lower boundary is when the thermistor is 400 ohms, and the upper when it is 3600 ohms. The upper boundary is a digital count of 0 as the internal DAC counts a descending voltage, and the lower boundary is a count of 128.

The scaling coefficients chosen for the recommended thermistors are the nominal calculated values from the manufacturers data. For cost reasons the tolerance is such that some user calibration may be necessary with some thermistors. This is easy with the aid of a jug of warm water, and the domestic fridge, together with a thermometer. Use the computer directly, rather than a multimeter, the coefficients are midrange offset and slope (see later).

R3 (1K2) and Zener diode D2 (6.2v) are protection components to protect the TL507 from impulse voltages on the long sensor wiring.

Terminal 12 on the relay connections is connected to 0 volt and it is recommended that this is attached to Mains Earth.

Since there is 240 volts on the printed panel it is also suggested that the board is defluxed with methylated spirits and then given a generous coating of polyurethane varnish over the relay copper conductors, to prevent surface conduction after a time, when dirt and moisture can ruin what was a satisfactory circuit insulation.

The layout of the printed panel is shown in figure 3. Construction is conventional, and there are only a few links to be made with cut off resistor ends. The ribbon cable needs care, especially at the ZX81 connector, which needs to be female to male to enable RAM packs to be fitted. Those with their extra RAM within the ZX81 case may smile with the author.

Software

The computer has to have at least 4K of memory. To run in 4K the display file has to be fooled into not padding out a full screen by the common trick of POKE 16389,76. Do not NEW after this or the ZX81 will indeed reduce to thinking it only has 3K. Apart from this there is no other system disadvantage, and the 3K static RAM additions were more reliable than the early 16K RAM packs.

The program is entirely in BASIC and SLOW mode is necessary to run the real time clock. FAST mode is set for the CALENDAR entry routine, when the clock is lost anyway, to speed things up! The real time clock has to be trimmed to suit the tolerances of the ceramic filter used in the ZX81 instead of a Quartz crystal. This is done in BASIC. The cycle of reading the three sensors, calculating the optimum start times, setting the output relays, reading the internal clock, resetting it, checking for midnight, calculating hours and minutes, updating the display, checking for keyboard entries, and starting all over again takes up to a minute, depending on the sensor readings. This leisurely pace keeps the ZX81 quite busy and is quite adequate for the task! It is left as an exercise for the reader to produce a machine code version retaining the floating point arithmetic.

Extensive use is made of well known ZX81 tricks to save bytes. In particular the variable I is set to 1, hence NOT I is zero. PI as an array subscript interprets as 3.

The line numbering is arranged in groups to identify the major activities:-

10	SLOW mode after keyboard entries after line 8000. SLOW is necessary to enable the frame counter clock.
20	Select sensor
30	Code relay states
40	Reset ADC
50	Count clocks to ADC
60	Clock inactive to ADC
70	Clock active to ADC
80	If ADC does not call halt then next.count

90	Convert count into logarithmic ratio
110	Obtain Room temp
120	Obtain Water temp
130	Obtain External temp
1000	Default zone stop time midnight (432E4 frames)
1010	Set room early to zero
1020	Set water early to zero
1030	If in zone 7 then do not calculate Early times
1040	Set room early to 2E5 frames (66min) in case time would be infinite
1050	Decode calendar hrs mins of next zone FROM time, which is the current zones stop time, in frames
1060	Calculate max possible room temp at this External temp
1070	If next zone set point less than max, and current room temp is less than max then calculate Early time
1080	Calculate Water Early time (Linear term only due to the transfer function of domestic tanks)
1500	Choose room set point, are we in Early start?
1510	Choose water set point, are we in Early start?
1520/30	Convert Early times to minutes
1540	Increment zone number?
2000	Room demand?
2010	Water demand?
2020	OR is logical (Room OR Water)
2030	If Room demand set up extension time (1E4 frames, 3.3mins)
2040	Put room back on if extension not cleared
3000/10	Read MSB of frames counter/LSB
3020	Update frames record (frame counters decrement)
4000	Is MSB still adequately 'wound up'?
4010	If LSB near zero reread in case of ambiguity during transition
4020	Rewind frames counter (less bytes than POKE's)
4030/40	Update Clock value/clear frames record
5000/10	Midnight?/Subtract 432E4 (24 hrs) from clock
5020/30	Clear extension Room demand/Zone 1
5040	Increment day,
6000/10/20	Time=clock+frames record/convert Time to Hrs mins
7000-7170	Print display (7110 is total RAM used monitor)
8000/10	If key A (STOP) pressed set flag S to 0/Check S
8020/30	FAST mode/Display STOP flag
8040/50/60/70	Set Clock=Time/clear E, F/Pause until key pressed
8080	Check for Continue, Set flag S
8090	If Continue set zone to 1
8100/10	If key D increment Day/If key Z increment zone
8120/30	If key F overprint calendar entry, input value, not in zone 1
8140	Take absolute value with 2 decimal places if From time < 24 hrs
8150/80	If key R overprint calendar, input value/Key W the same
8190/00	If key T overprint time, input value in HH.MM
8210	If T has been changed then it is HH.MM. Code to frames, traps
8220	If key S then SAVE 'P' to tape recorder-Autorun on LOAD
9000	Initialising. First entry to program is RUN 9000. Subsequently use GOTO 1 to retain calendar data

Coefficients

Sensors are rounded to 1 decimal place. In lines 110, 120, 130 the second number is the offset in degrees C★10. The first number is roughly equal to the half range★10. i.e. the room sensor has a mid range value of 12°C and a span of 16°C approx up and down, 32° in all.

Early calculation time constants in frames are set at 8E4 and 3E3 for room and water respectively, in lines 1070 and 1080.

Room heating capacity in line 1060 assumes Water flow temp in radiators of 70°C and ratio of Gain and loss coefficients of 3 (INT PI).

Real time clock trim: Alter 65535 in line 3020.

Program Entry, and operation

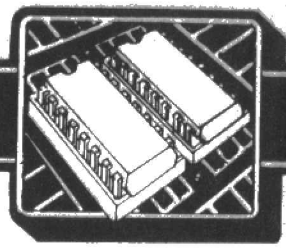
It is easier to practice with the program without the controller connected, and the operation studied.

Type in the program in the normal way, but 4K owners must enter POKE 16389,76 to unpad the display (see manual page 177), do not NEW. Note that 'STOP' in line 8030 is shifted A, and ★★ is shifted H in lines 8120, 8150, 8170, 8190.

To run use RUN 9000 to declare the arrays and the initial variables. Within about 5 seconds the display is built, and the ZX81 pauses in Fast mode. Notice that STOP is printed over the top left ZX ENERGY logo to indicate that control has stopped, as has the clock.

Keyboard entries may now be made to set up the calendar etc:-

- D will change the current DAY
- Z Will change the current time ZONE
- F will overwrite ★★ at the current FROM time, and allow time entry in 24 hour notation HH.MM (nb: zone 1 is locked as zone 1 must start at midnight)
- R will overwrite ★★ at the appropriate ROOM set paint, and allow numerical entry. Decimal values are accepted and used but not displayed. Nonsensical entries are not trapped.



W as R for WATER set point.
T will overwrite ★.★ at the time display. entry as for F.
S will SAVe program and all data to tape, permitting autorun on tape loading. (4k owners to POKE 16389,76 before Loading).
C will CONTINUE and run the controller, in SLOW mode, and remove the STOP flag.
A will STOP the controller when running and enter the keyboard entry routines above. Key only checked once per control cycle (20-40 secs).

If the program is halted with the BREAK key, GOTO 1 will resume without loss of data. CONTINUE may yield an incorrectly formatted display.

Display

The Day, Time and Temperatures are shown. The set points currently being worked to are shown alongside the relevant temperature. A black arrow pointing at a set point shows that it is demanding heat.

The Room demand has an Extension timer which affects the Room demand relay only, not the 'OR' relay. This scheme allows for a circulation pump to run on after the boiler has stopped firing, to extract the residual heat. The black arrows show the status of the Room and Water relays only, and the extension time may be seen. The extension time is rounded up to the next occasion that the ZX81 services that program line.

The Early times show the results of the optimum start calculations. The advance is necessary to achieve the next time zone set point at the zone start time. Optimum start is not made on zone 1 at midnight. When an early start has commenced the next zones set point is displayed. If the warm up is too rapid the controller can revert to the previous zone for a while. This action would indicate that the calculation coefficients do not match the heating equipment capacity. The coefficients are chosen to give 61 minutes room heating advance when room and external temperatures are 0 degC, and the set point is 21 degC (70F). The same applies to the Water, 0 degC aiming at 60 degC in 60 minutes. (In fact the sensor is scaled to 31.9 degC min which gives 28 minutes). These times may be rescaled to suit local conditions.

As written the program will return sensor temperatures — 5.3 31.9 —5.3 degC respectively as the minimum values. By setting K equal to 8192 the values 29.5 88.5 29.5 degC are the maximum values. Without the controller the program defaults to reading the ROM at addresses 0 and 2, and the contents there give the appropriate replies as if from the ADC.

The total RAM used by system variables, program, display and variables is displayed for development purposes. As written the value is 4306 bytes. With the display not padded out with spaces by entering POKE 16389,76, the value is only 3832, but this will rise up to 3845 bytes when screen entries are made to the calendar etc. However with 4096 bytes available, there is still around 250 bytes available for the Z80 stack etc., more than adequate.

Users really desperate for memory space may delete lines 9000 to the end after initial running. So long as only GOTO 1 is used or autorun from tape, all is OK, (RUN will delete your laboriously entered Calendar), this trick will release 200 bytes.

Perspective

The scale of this project goes much further than the building of the ZX ENERGY controller. The task of implementing the system into the house in a workman like fashion, and tracking down and suppressing the Heating equipment so that the system will run reliably and safely is the major effect, and must not be underestimated.

The constructor who perseveres and plans his own implementation of this system into his heating, and completes the project, will derive great satisfaction from actually doing something that has a Real World use for a computer, rather than another 'invented' application. For instance the computer personal finance systems are great fun, but pencil and paper are easier.

As a control machine the ZX81 has been demonstrated with robot mice etc., but on your heating it is not unreasonable to try to save money on your energy bills, and recoup the cost of the ZX81.

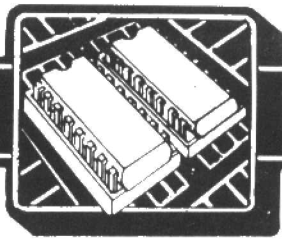
The control method is the simplest on/off and an error in temperature has to build up before the system responds. There is processing power available in the ZX81 to implement some of the more advanced control algorithms known as Proportional control with Integral and Derivative action. If interest is sustained this may be published as an upgrade, but in the meantime the reader will have to resort to the public library. The PID algorithm can achieve a far smoother and almost unchanging temperature.

Now can you get your Energy Management System going before the winter ends? Good Luck.

CONTENTS

	Quan.
IC1 74LS273	1
IC2 74LS244	1
IC3 74LS32	1
IC4 74LS00	1
IC5 7403	1
IC6 TL507C	1
IC7 7438	1
D2, D3, D4, D5 IN4148	4
D2 BZY88 C6v2 Zener	1
R1 470 ohms ¼ watt	1
R2, 3 & 5 1K2 ohm ¼ watt	3
R4 2K2 ohm ¼ watt	1
C1 & C2 10μF 16v	2
Thermistor VA1039 Mullard 500 ohm	2
Thermistor VA1109 Mullard 4K7 ohm	1
LED 0.2" Red	1
Contact Suppressors	3
Ribbon Cable 9 inches 20 way	
Plug 2 x 23 way	1
Socket 2 x 23 way	1
Terminals 90°	3
Terminals 45°	1
RL1, 2, 3 Relays 12v 400 ohm coil	3
Plastic Box 7.8 x 14.5 x 4.5 cm	1

Construction details continued on Page 86.



High Resolution Graphics

by John Chewter

This special feature on High Resolution Graphics forms part of a series of articles started in October 1982, the object being to build a state-of-the-art computer at a price considered to be low when compared to a small business system of comparable specification. The most important feature on the computer system is the high resolution graphics capability, of which full details are given in this month's issue. New readers who would like to receive back copies of the whole series to date should write to E & CM, Back Issues, 40-42 Oxford Street, Daventry, Northants.

Address Decoder (Fig. 1)

The heart of this circuit is the 3 to 8 line decoder IC45 (74LS138). To enable this device it is necessary to pull pins 4 and 5 down at the same time.

Thus the decoder is enabled for any address starting with F1YX (Where Y is a number less than 8 X is any hex number).

Bus lines 25, 26, and 27 are fed to the A, B and C inputs of IC45, which means that Y∅ represents F1∅X, YI = F11X etc. Thus, VDGE has a low going pulse on it for any valid address on the address bus from F1∅∅ – F1∅F (This signal enables the video display generator chip and the least significant four bits determine which register is addressed).

COL produces its signal for address F12∅ – F12F and as it is gated with RW, only occurs for a write signal (this is for the colour register).

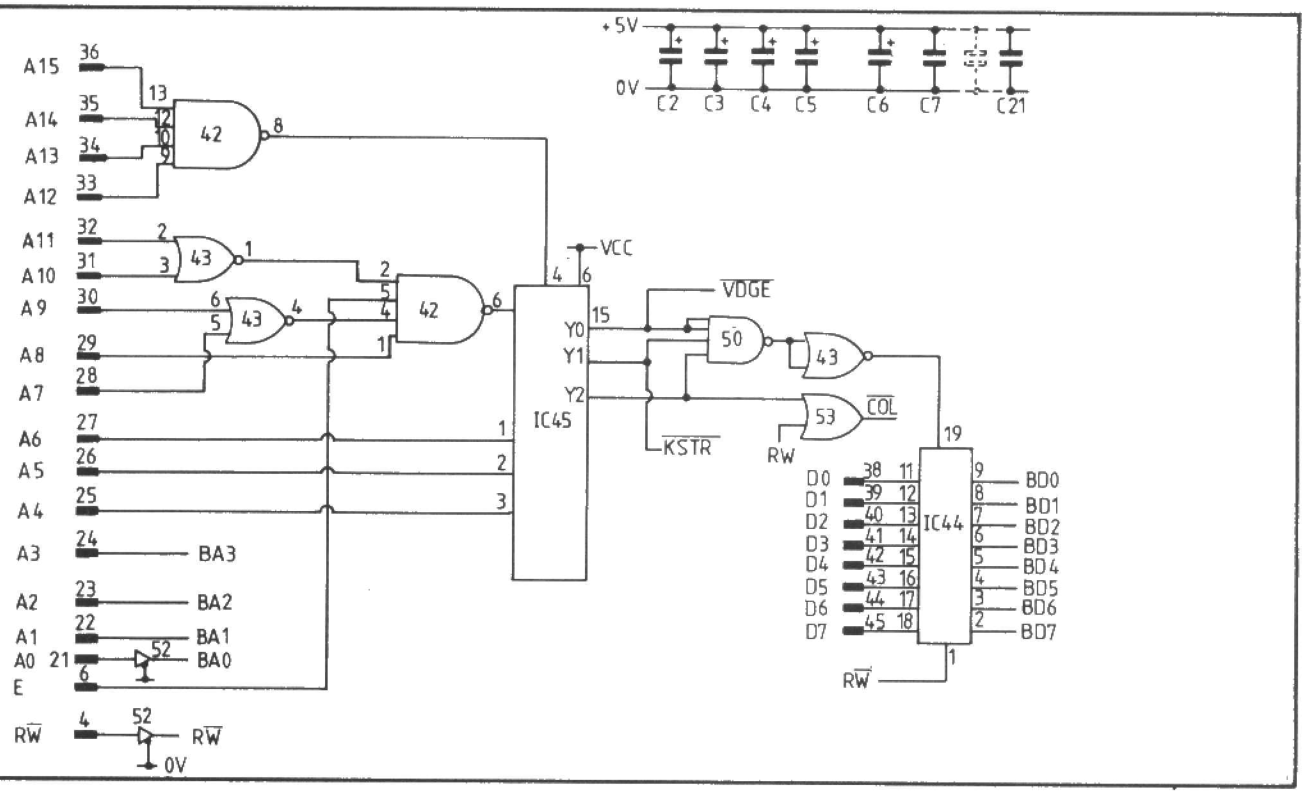
KSTR responds to F11∅ – F11F (read or write) and is further decoded in fig. 4.

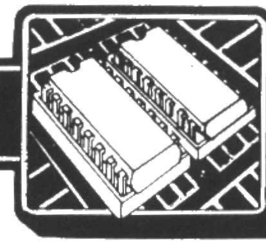
IC44 is the board's data buffer and is enabled (pin 19) for any address F1∅∅ – F12F. The direction of this bi-directional buffer is controlled by the RW line all bus connections present 1 LS TTL load to the bus.

Testing the decoder, using S-Bug, is fairly simple.

Using the 'Q' command (memory test) enter the following F1∅∅ – F1∅∅. This performs the memory test on F1∅∅ **only** and thus produces a stream of pulses, hopefully, from VDGE. (The test will produce error messages which are to be ignored). The pulse stream is about 1 pulse/sec. for a 1200 baud terminal and will vary according to baud rate i.e. it produces a pulse after each error message (actually, it is 2 pulses very close together). To observe these pulses, we found that it was necessary to set the 'scope time-base to its slowest scanning rate, when the pulses were clearly visible. **Warning** Stopping the time-base can permanently damage the display tube. Repeating the test for F12∅ will exercise COL and F11∅ for KSTR.

Fig. 1





SEE PAGE 66

Timing Generator Fig. 2

IC7 and associated circuitry composes a 14MHz clock generator. Please use a 14.00MHz crystal (would you believe that I was told that no-one makes them (!) and that some other value that they had in stock was near enough – it wasn't!).

The 14MHz Clock is fed to IC1 which is configured as a three bit syncronous counter. Q0, Q1 and Q2 represent the current count as a three bit code (i.e. it counts from zero to seven and repeats itself).

When this is decoded by IC4, which is a 3 to 8 line decoder, we have eight lines, normally high, which are pulled low and released, sequentially. To obtain the RAS signal (which needs to be high for YØ and Y7 periods) we feed YØ and Y7 into a nand gate, the output of which is normally low, because during phases Y2 – Y6 both inputs to the gate are high.

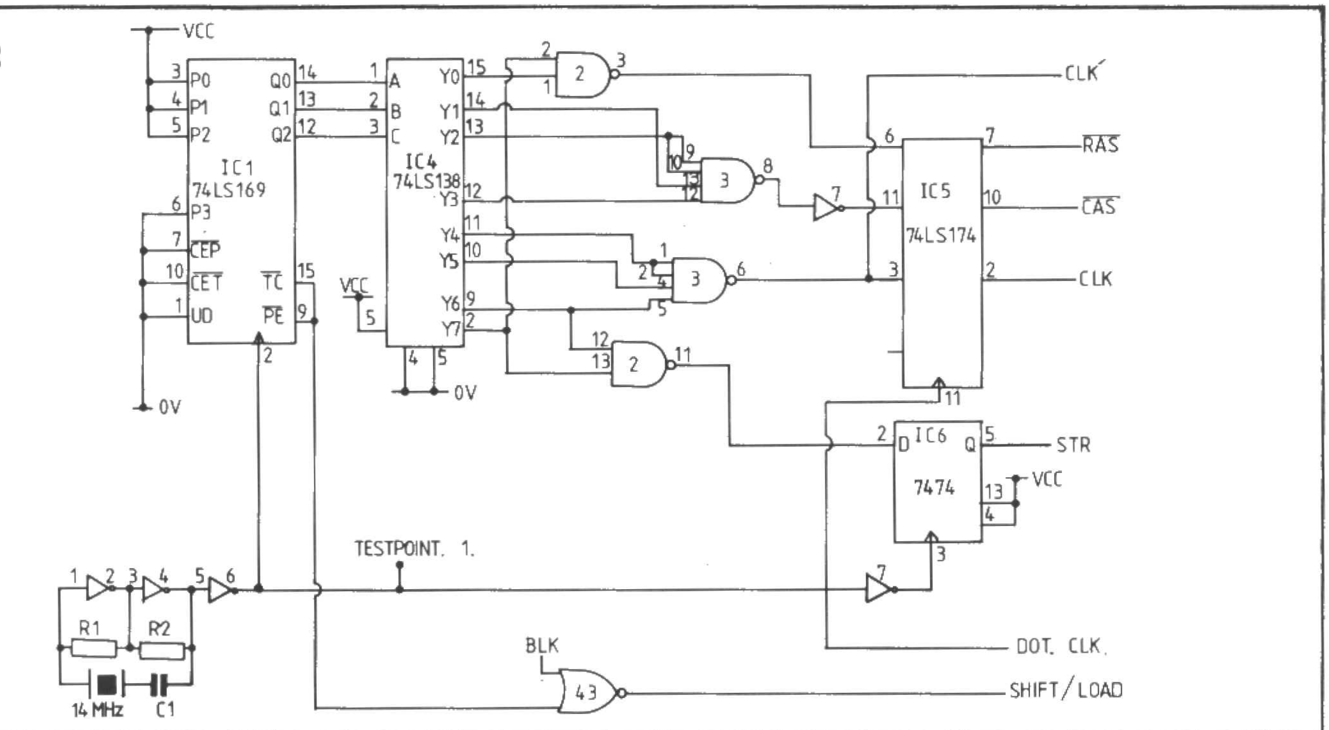
When either Y1 or YØ goes low then, the gate output goes high for this period thus producing the RAS signal, which is latched in IC5.

The other signals are produced by similar means. The STR signal is latched by the antiphase of the 14MHz clock because if we did not, data would be clocked to or from ram just as its address became invalid.

CLK is taken before the latch to compensate for gate delays in the page switch (fig. 3). The shift/load signal is derived from the carry output of the counter.

Testing the timing generator is simply a matter of ensuring that the wave form is as per fig. 6.

Fig. 2



Video Generator Fig. 3

The main integrated circuit here is IC37, the EF9365. The IRQ signal (pin 13) is buffered and brought to pad 'L'. As previously stated, we have avoided interrupts wherever possible, however, there will, no doubt, be people who wish to use this board with other systems or configurations.

To connect the IRQ to the backplane link L and K. **Be warned.** Connecting this in the standard system will cause a system crash because there is no interrupt handling routine in the standard firmware, that will clear the source of the interrupt.

We drive the board by polling. Brave individuals may, of course, write their own interrupt handler. (If we had not incorporated this option we bet that poor old Ted would have a pile of letters next week!).

IC38 buffers the address lines from the VDG. The outputs are fed via. 22ohm matching resistors to the ram array.

MSLØ, MSL1, MSL2, MSL3, ALL, DIN, DW and BLK are latched into IC39 by STR so that these signals remain available to the system for the entire CLK cycle. IC40 is a clever 3 to 8 line decoder – with "bells & whistles" which enable most of the RAS signal processing to be carried out, using only one extra gate.

The complete functioning of this part of the circuit is more complicated than the diagram would infer.

During normal display cycles RASØ to RAS7 follow RAS because all eight rams (per plane) need to feed the shift registers simultaneously – however – during write cycles the rams are addressed singly (per plane).

Testing this part of the circuit could be a little tricky because although display cycles can easily be seen on a 'scope, the write cycles happen so quickly. We have had no problems in this area (so far . . .).

The sync signals are merely buffered and inverted and fed to the video connector. They are pulled-up to VCC to give a 5 volt output.

TR1 constitutes a mixer that combines the video and sync signals, on the GREEN channel only, so that a single memory plane (monochrome) display may be produced. When building a monochrome display, it is important that the border "patches" are set to "black" (G-A, H-B, J-C).

Only the green channel memory devices need be fitted, of course. When using the composite video output, the monitor should be terminated with a 75Ω resistor (i.e. at the monitor end of the cable it is standard practice to connect a 75Ω resistor – not wire-wound – across the two conductors) if your monitor says 75Ω or similar, it is probably terminated. Anyway, try it with or without if you are not sure, and use the option with the best results.

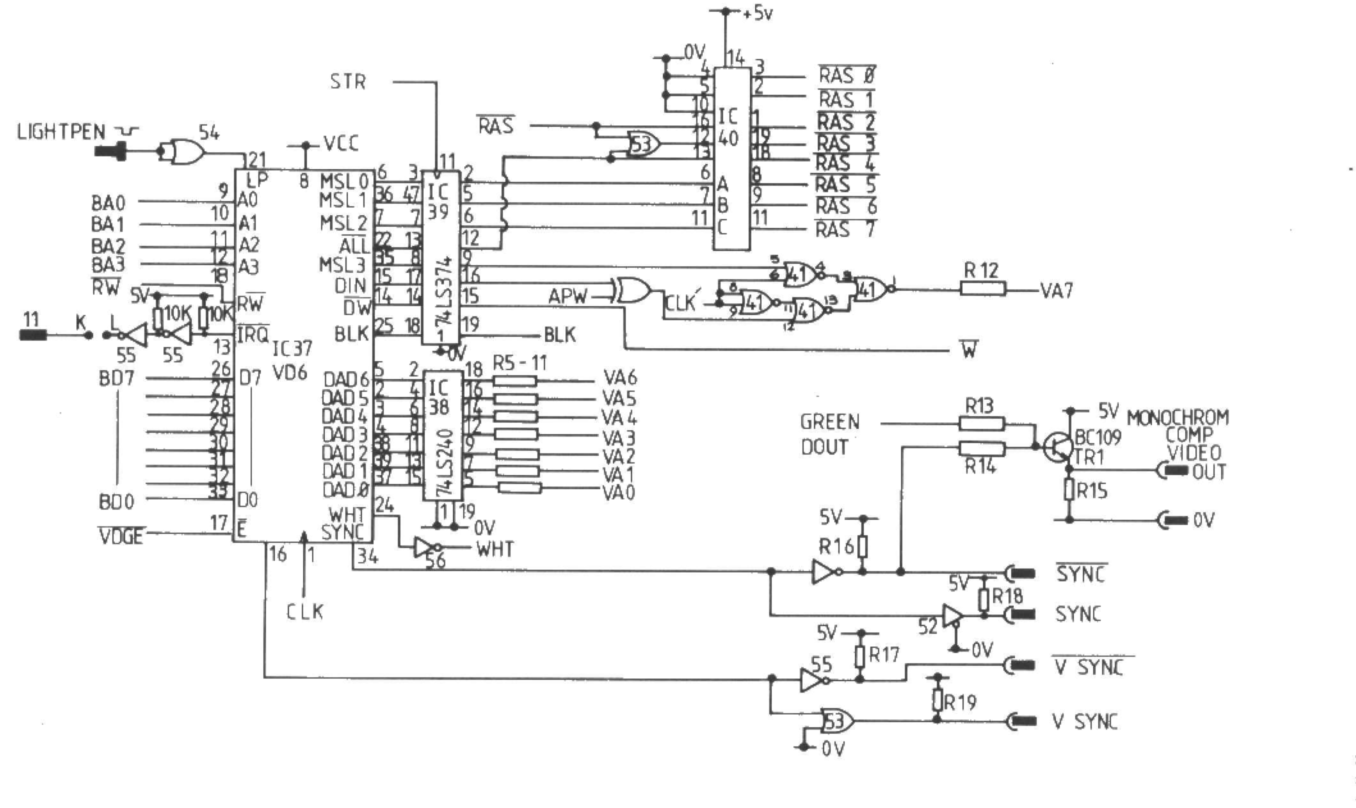
Connecting to a U.H.F. modulator will be covered in a future article. IC41 comprises a poor man's data selector. As by now you will be aware the address for the dynamic rams is multiplexed according to the states of the RAS and CAS lines.

The data selector selects either MSL3 address line or the DIN signal from the VDG (which now becomes the page select line). These two signals, multiplexed together become address bit seven and are fed via. the matching resistor to the ram array.

The page select line IC39 pin 16 may be inverted by asserting the APW line (alternate page write). This signal derives from fig. 4. Whilst in normal mode, not write-only mode, the BLK signal is asserted (high) when the VDG is not in the display area.

Writing to screen takes place during the BLK periods only, then, if this signal is exclusively or'd with the page select line, then its inverse results. i.e. if page one is selected then page two will be written to. This enables such tricks as assembling complicated graphics invisibly, while the operator is reading say, the next instruction, and then surprising him by instantly presenting him with a screen-full or it is useful for animations, as the next frame may be drawn and then alternated with the first picture, to apparently speed up the drawing etc. etc.

Fig. 3



Keyboard and Colour Control

The colour register is fairly simple. The register is a write only register which resides at F12Ø. The Red, Green and blue Din signals set the "data in" lines to the RAM chips, as described last month.

The alternate page write line is a new addition however and responds to bit 3 of the register – a high to enable APW operation.

IC48 is the read – back register – not previously described. This appears as bits 1, 2 and 3 or F11Ø with bit 1 = Blue, 2 = Green and 3 = Red. The function of this part of the F11Ø register is to store the colour combination of the last write cycle (or 'supposed' write cycle) to screen ram – see later.

Bit Ø of F11Ø is a status bit which is set whenever a key is depressed on the keyboard. It is cleared when F111 is

read. (This is the address of IC46 – the keyboard data latch). This combination simulates an ACIA to the standard software that is available.

Data is latched into IC46 on the rising edge of KBD – the keyboard strobe, thus it will be seen that the keyboard needs to have a negative going strobe pulse (not a level).

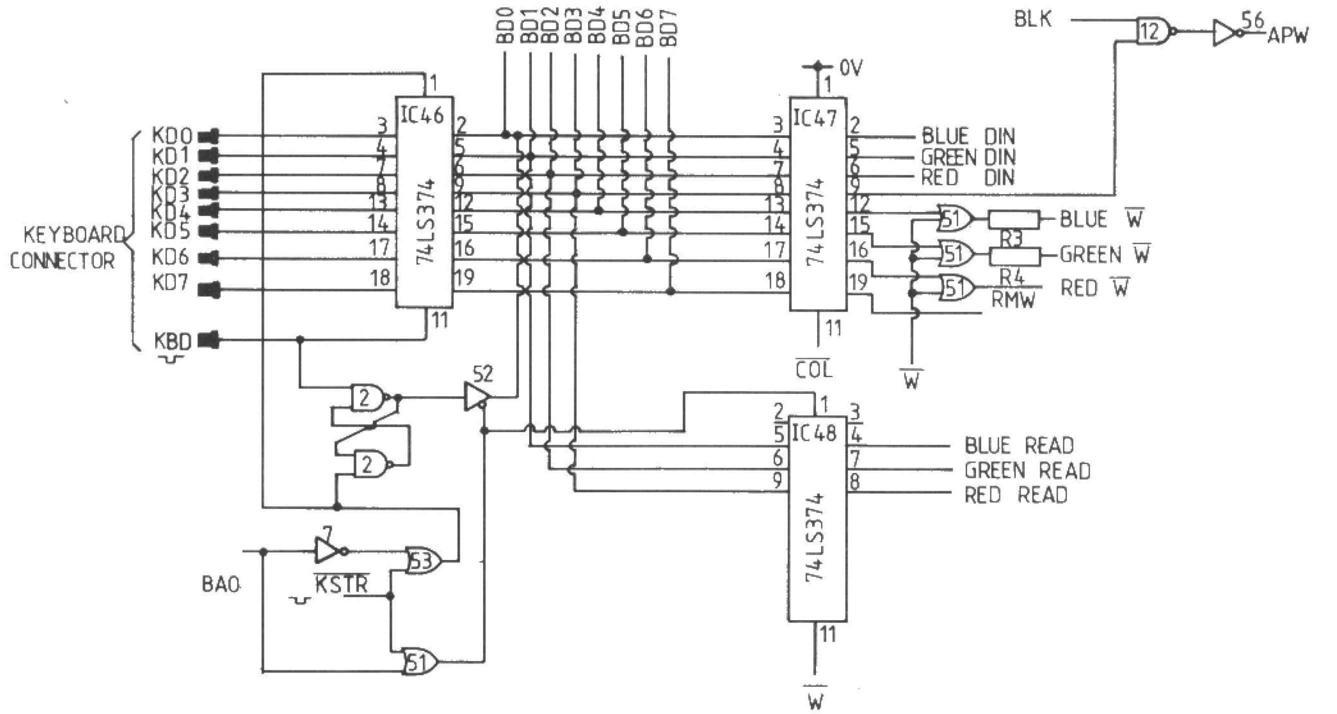
We have been using a Carter Associates keyboard type 756MF. This is the metal frame option which we found to be well worth the little extra – it is very solid and has an excellent feel to it.

It does require – 12V but a small DC/DC converter may be mounted on the board – type DC – 512 which means that the keyboard can be operated from +5V only. At present, the only place that —12V is used on the system is RS232 interface and if a 27Ø8 is to be used on the C.P.U. card (unlikely).

Thus if RS232 is not to be used in the final system then the entire system can be driven from a +5 volt supply. The +12V supply would also not be needed.

A case is also available, for the keyboard, and its part is CAR/20. The keyboard was obtained from Watford Electronics (address as per their advert).

Fig. 4



Ram Array and Output Fig. 5 refers

To avoid duplication and repartition we are only going to discuss the operation of one array viz Red.

The other two are identical apart from the composite video output being connected to the Green output (Fig. 3).

Where, say, Red W is shown then obviously Blue W would be the Blue array's equivalent etc.

The outputs of the RAMS are fed to the 74LS166's syncronised by the shift/load signal and the byte is then shifted out by the dot clock and thus becomes Red Dout when Or'd with WHT.

WHT. goes high when a light pen read sequence is intiated by a "Ø 9" command. This forces the screen to white so that anywhere on the screen may be detected. The command would need to be initiated by the switch in the light pen.

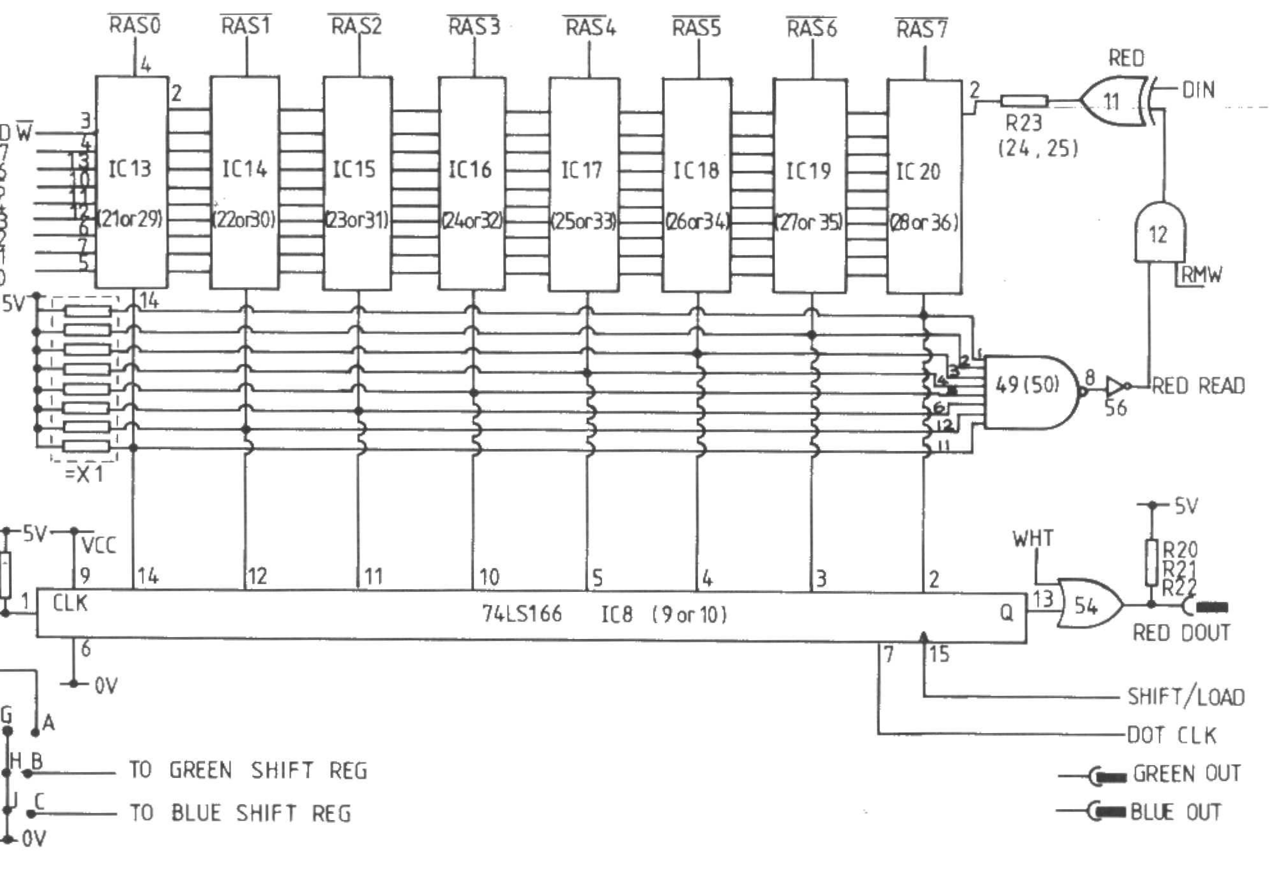
The L.P. strobe (low going) latches the position in the L.P. registers on the positive going edge.

The colour of the border area is determined by the serial inputs (pin 1 74LS166). The shift/load line loads the shift registers with ram data and then shifts out the next 8 pixels. However, when outside the display area, this signal disappears – but the dot clock continues. If we grounded the serial inputs then Ø's would be shifted out, hence producing a black border. This would continue until the next shift/load signal on the next line. Connecting G-A, H-B and J-C will give a black border. Not making any of these will produce a white border and making some, and not others, will produce coloured borders. If A-D, B-E and C-F are linked, then the border is now under software control, because the border control is now connected to the three least significant bits of the CPU card output port. Loading this latch (EØØØ) with the same codes as for the

colour register will produce a border of the appropriate colour.

The other feature, in this part of the circuit, is the read- modify-write facility. When we are about to write a pixel, to a given address, the data already contained will be presented on one of the data lines (pin 14) of the DRAMS. Non selected ram chips will be tri-stated off and so will produce data "highs", because they are pulled up by the 10K pull-ups. If the selected output is low, then one of the inputs to IC49 (or 50) will be low. This will produce a low on the output IC56. If RMW is high then this will and'ed to produce a low which will not invert DIN and therefore a high will be stored. The complement of the colour previously stored is now displayed. If the selected ram's output had been high then this would have produced a high at the output of IC12 and, thus, inverting DIN to store a low. Of course it is necessary to insure that the DIN's are high before performing this function.

Fig. 5



Hi-Res Graphics Board Parts List

INTEGRATED CIRCUITS

IC1	74LS169
IC2	74LS00
IC3, 42	74LS20
IC49, 50	74LS30
IC4, 45	74LS138
IC5	74LS174
IC6	74LS74
IC7	74LS04
IC8, 9, 10	74LS166
IC11	74S86
IC12	74S08
IC13-36	HM4864 – 2 (PLASTIC) or equiv.
IC37	EF9365
IC38	74LS240
IC39, 46	74LS374
IC47, 48	74LS374
IC40	AM 25LS2538
IC41	74LS28
IC43	74LS02
IC44	74LS245
IC51, 53, 54	74LS32
IC52	74LS125
IC55	74LS05
IC56	74S04

RESISTORS

X1	10K RESISTOR PACK
R1, 2, 13, 15	1KO ⅛W
R3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 25	22R ⅛W
R14	680R ⅛W
R16, 17, 18, 19, 20, 21, 22, 23, 24, 25	560R ⅛W

CRYSTALS

14.00MHz (NEAR EQUIVS WILL **NOT** WORK CORRECTLY)

CAPACITORS

C1	10pF
C2, 3, 4, 5	10μF 16V TANT CAP
C6	100μF 16V
C7-C21	0.01μF decoupling types

TRANSISTOR

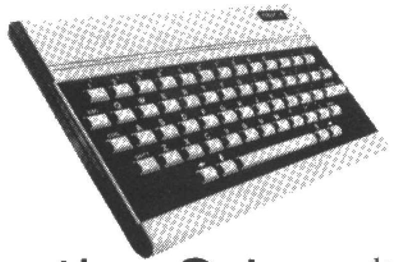
BC109

KEYBOARD MODIFICATION

74L00 1 off
4K7 Resistor 1 off
74L00 1 off
4K7 ohm 1 off

Teddy Technical
continued on Page 91.

THE ORIC I



by Mike James

There are quite a few small, low cost micros on the market at the moment and the Oric I is certainly one of the least expensive. If you are thinking of buying something like a Spectrum or a Dragon then should take a look at the details of the Oric before you buy. If you have already taken the plungs and bought a micro, (and we all have to commit ourselves sometimes!) then it is worth looking at the Oric to see if it shows up any shortcomings of your chosen machine.

If you were given the job of designing a low cost personal computer then to make it successful you would have to try to improve on the features of the existing machines or try to make it something completely different. As you might imagine it is difficult to make a new computer very different from all those that have gone before and so most machine designers have to content themselves with making improvements, and in the small computer field this means more hardware for less money. The Americans have a quaint expression – "more bang per buck" – which sums up the small computer market very nicely. So the question that has to be answered about the Oric is does it provide more "bang per buck" than the competition?

It is difficult to talk about the Oric without continually comparing it with the Spectrum. The simple reason for this is that of all its competitors, the Oric is superficially more like the Spectrum than any other. I say superficially because to assume any direct comparison would be to miss the fact that the Oric is a very different machine. Indeed, many of the Oric's features can best be described by explaining how they are different from the Spectrum's. For starters, it uses a 6502 microprocessor whereas the Spectrum uses a Z80. On the software side it uses a modified Microsoft BASIC whereas the Spectrum uses its very own ZX BASIC. However all this is rushing ahead a little. Let's first examine the Oric itself, starting from the outside of the case.

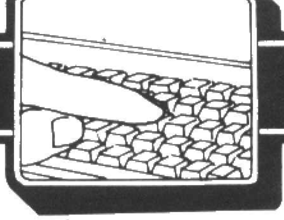
Using the Oric

The Oric is a rather smart looking machine in a plastic case that measures just seven inches by eleven inches. A novel feature is the use of a wedge shaped case to give a convenient angle to the keyboard. The keyboard itself is a sort of cross between the approach taken by the Spectrum and a standard keyboard. The actual keys are made of hard plastic and move but they can hardly be compared to the switches used in a 'real' keyboard. This is not to say that the Oric's keyboard isn't comprehensive – it has every key you could possibly want and a few more, but it somehow feels slow. You won't find any Spectrum or ZX81 style keywords on the keyboard and this means that each BASIC command has to be entered letter by letter. There is a lot to be said in favour of the single keypress keyword entry when you are learning BASIC but in the end it's all a matter of taste.

To get going all you have to do is connect the separate power supply, a TV set and (optionally) a tape recorder. (In which case you'll need to provide yourself with a cassette lead as there isn't one supplied with the Oric). There is no on/off switch and the Oric's display appears as soon as the power is connected and the TV is tuned in. You cannot avoid noticing the 'keyclick' that accompanies each keypress because it is a very loud, high pitched beep. Mercifully it is easy to turn this noice off because it very soon becomes irritating. The standard text screen is composed of 28 rows of 40 characters with both upper and lower case characters. The screen scrolls automatically when full and all in all it gives the feel of a full sized computer. In this mode the Oric can be programmed in a fairly standard form of Microsoft BASIC and this is one of its strong points. If you specifically want a computer to develop Microsoft programs then the Oric is a bargain! However, although there is the facility to save and load programs on tape there appears to be no way of saving data on tape so developing data processing programs might be a little difficult.

Graphics

Good graphics are a must for any small computer that hopes to be successful in the home micro market. The reason for this is that one of the main uses for this type of computer is in game playing. The Oric has two main display modes – text/lo-res and high-res graphics. This is again distinctly different from the Spectrum's use of a single display mode for both text and high resolution graphics. A second difference is the use of serial rather than parallel attributes. If you are trying to display a screen in colour there are two problems to be solved. Firstly, you have to specify what shape should be displayed at a particular screen location. This is essentially the same problem as displaying a character or graphics shape in black and white. Secondly, you have to specify what colours will be used to display the shape. This is essentially the problem of adding colour to a black and white display. The Spectrum and most other computers use a method of storing the colour information that is known as "parallel attributes". In this method the colour information is stored in its own special area of memory. The exact way that this is done varies from machine to machine but the key fact is that each screen location has information concerning its colours stored, even if it is the same colour as all its neighbours. The Oric doesn't use this method! Instead it uses the technique of serial attributes introduced and used successfully by teletext. The basic idea behind serial attributes is that you store a special code, an attribute code, in the screen area of memory and this effects the colours of all of the characters to its right and on the same line unless another attribute code is encountered. There are two important things to realise about serial attributes. The first is that you don't have to store attributes for each screen location, only the colour changes. The second is that



you do have to use up a standard screen location to store an attribute. Thus there are advantages and disadvantages of this method. In my opinion serial attributes are more trouble than they are worth but I am prepared to admit that this feeling might be based on the fact that I am more used to the parallel attribute method of the Spectrum. It is important to realise that both the parallel and the serial attributes method have their limitations and there is no substitute for a real colour graphics display where every dot on the screen has to be given a colour code.

The Oric's low resolution screen is 39 wide by 27 high but the effective resolution is higher in that you can use teletext graphics blocks within each graphics cell. In this mode you can also mix text and graphics in eight different colours, subject to the limitations implied by serial attributes. As all of the characters that the Oric can display are 'downloaded' into RAM when the machine is first switched on the entire character set can be redefined by the user. However, Oric BASIC doesn't provide any special commands to redefine a character so you have to resort to unaided POKEs.

The high resolution graphics mode provides 240 by 200 plotting points in any two of the eight colours. If you want to use all eight colours then you are reduced to an effective colour resolution of 40 blocks by 200 lines. In other words you can set or un-set any of the 240 by 200 points but only change colours every six horizontal dots. There are special BASIC commands for plotting dots, lines and circles. There are also two ways of printing characters on the high resolution screen. The bottom two lines of the screen are in fact a 'window' onto the text screen and can be printed on in the usual way. This is rather reminiscent of the APPLE II's high resolution screen but unlike the APPLE the Oric also provides a special function CHAR which will draw a character anywhere on the high resolution screen. This is better than nothing but I would much prefer a high resolution graphics method that allows the free and easy mixing of graphics and characters.

The Oric's graphics are good enough for playing games and many serious applications. If you need a low cost teletext or prestel graphics displya then the Oric might well be just the machine for you.

Sound

The second important feature for a successful low cost micro is its sound capabilities. In this area the Oric has a distinct advantage. Its sound capabilities are only surpassed by the BBC micro which is, of course, a more expensive machine altogether. There are a number of predefined noises such as ZAP, EXPLODE etc. that are just right for writing games programs. There is a SOUND command for producing the odd beeps etc. and there is a MUSIC command to help you translate music to play on your Oric and last but not least a PLAY command that provides a limited but very easy to use envelope facility to make non-standard special effects. As well as noise channels there are three separate tone channels so three note chords are possible on an Oric! Before moving on to other details it is worth mentioning the fact that the sound that issues from an Oric is very loud and very clear.

The Hardware

To get inside an Oric you have to invalidate its guarantee because the final fastening screw is hidden under a well stuck down metal strip which clearly warns of this consequence. If you do get inside, however, you will be greeted by a well made machine. All the components are contained on a single printed circuit board. The keyboard is held on a second printed circuit board under the main board. As already mentioned the heart of the Oric is a 6502 but there are three other large chips on the board. A 6522 VIA is presumably used to interface the keyboard, the centronics printer interface and possibly other odds and ends. The presence of a 6522 makes it all the more surprising that the Oric seems to lack a timer function. The 6522 certainly has enough inside to make a timer possible so perhaps the Oric secretly already has such a function or will have one in the future. The second large chip seems to be a custom-made device or ULA. The advent of the ULA is the single advance that has made the sort of small micro typified by the Oric possible. Without the ULA the Oric would have to have had 50 or 60 standard TTL chips at much higher cost to achieve the same result. The final large chip is the well known AY-3-8912 sound effects generator. The RAM (in the 48K version at least) takes the form of eight 4164 chips and there is no room for any extra memory expansion on the main board – mind you its difficult to see why any should be necessary!! The BASIC etc. is held in a pair of 2764 EPROMs. Whether this is a temporary measure to be replaced by a pair of ROMS, or even a single ROM, is something only time will tell. Finally, the subject of hardware cannot be left without mentioning the large (3") loudspeaker that is mounted in the middle of the printed circuit board – now we know where all that noise comes from!!

The hardware design and construction of the Oric is very good. It is nice to see that it is possible to do clever things with a 6502 instead of the usual Z80! However, with the presence of the 6522 and the sound effects chip it is surprising that a few extras such as a joystick input and timer are missing. These extras would have added very little hardware to the machine as they could have been realised mostly in software.

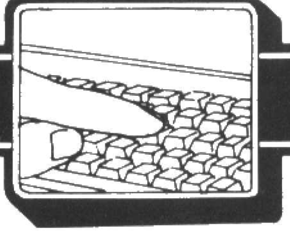
Documentation

When the review Oric arrived it only had a provisional manual. Fortunately the real thing arrived a day or two later. The real manual is well written and fine for learning BASIC and the basics of the Oric but there is a real need for another more advanced manual to help users get the best from their machine without having to experiment. It is a well known problem that documenting a micro intended for popular use is difficult. The main difficulty is finding the correct level – most manuals are either too simple or too complicated. The Oric manual strikes the right balance for an intelligent and inquisitive beginner but fails to communicate enough about how the machine is organised to enable any creative use of the machine. The only real technical detail to be found in the manual can be seen in the memory map (fig. 1)!

Expansion

One of the attractive things about the Oric is the range of expansion that is available or promised. There is a modem interface that should be useful for prestel

Continued on Page 50



Continued from Page 47

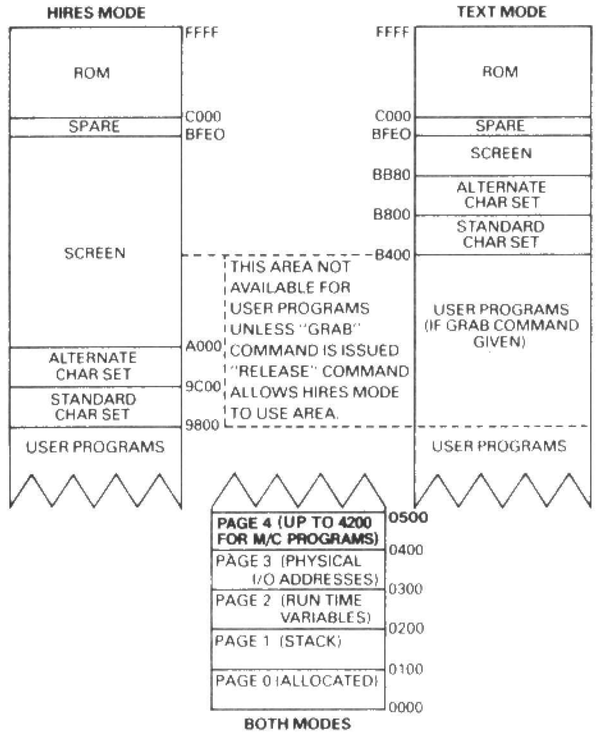
and other similar services. Promised expansion includes a disc unit and a special printer. The Oric with its tele-text/prestel graphics and a modem opens up new areas in affordable computing. However, there are some notable omissions from the advertised list of promises. For example there is no mention of any program cartridges, joysticks or speech synthesisers.

Conclusion

The Oric is a difficult machine to come to a definite conclusion about mainly because the manual doesn't give enogh information for a reviewer to be sure that every possibility has been explored. It has some features that are better than its nearest rivals, for example its sound. On the other hand it has some features that are difficult to assess against its competitors, for example its use of serial attributes in graphics. If there is a particular feature that the Oric possesses that you feel is important for your application, e.g. its keyboard or Microsoft BASIC, then it is a machine to consider carefully. Finally, the Oric's name was inspired by the isolent computer hero of the TV series Blakes Seven – Orac. I doubt that Oric is as powerful as its legendary namesake but then what do you want for a reasonable price?!

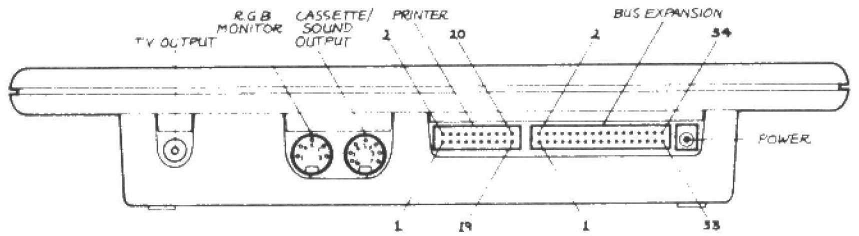
Figure 1

ORIC 1 MEMORY MAP (40K)



N.B.
1) For 16K systems all addresses (except ROM) are minus 8000 (hex)
2) All addresses are given in Hexadecimal

Reprinted from Oric Manual



Pin Output Chart

BUS EXPANSION

MAP	1	2	ROMDIS
ØZ	3	4	RESET
I/O	5	6	I/O Control
R/W	7	8	IRQ
D2	9	1Ø	DØ
A3	11	12	D1
AØ	13	14	D6
A1	15	16	D3
A2	17	18	D4
D5	19	2Ø	A4
A5	21	22	D7
A6	23	24	A15
A7	25	26	A14
A8	27	28	A13
A9	29	3Ø	A12
A1Ø	31	32	A11
+5V	33	34	GND

PRINTER

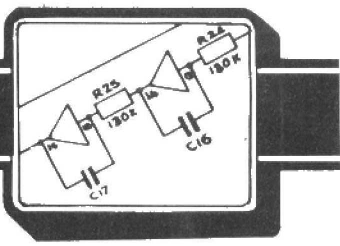
STB	1	2	GND
DO	3	4	GND
D1	5	6	GND
D2	7	8	GND
D3	9	1Ø	GND
D4	11	12	GND
D5	13	14	GND
D6	15	16	GND
D7	17	18	GND
ACK	19	2Ø	GND

R.G.B.
1 — RED
2 — GREEN
3 — BLUE
4 — SYNC
5 — GND

CASSETTE/SOUND
1 Tape Out
2 GND
3 Tape In
4 Sound 8912
5 Sound
6 Relay Contact
7 Relay Contact

Not referred to in article but shows potential for expansion.

151



Continued from Page 30

You will notice that the C0 line selects between AND and OR and that the C1 line inverts the result. A logical implementation is shown in fig. (9) and a workable circuit using NAND gates and inverters only is shown in fig. (10). I have called is the general purpose gate because it can perform four different functions. The only difference between this and an ALU is that the ALU has wider data buses and can select between more functions. In addition, of course, any ALU worth its name can also ADD and SUBTRACT. Lets see how addition is carried out on two 1-bit binary numbers. The truth table is shown below:

A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

It will be seen that the SUM of A and B is none other than the exclusive-OR function described last month, while the carry bit is simply A AND B. A circuit which produces a Sum and Carry from two inputs is called a half adder and one realization is shown in fig. (11). It is usually necessary, however, to add in the Carry from the previous stage. This requires a second half adder. A little thought will convince you that it is not possible for both half adders to procude carry's simultaneously so the two carries can be OR'ed together as shown in fig. (12). I will invite the reader to design an implementation of this circuit using NAND gates and inverters. Subtraction is often carried out by complementing one of the inputs first and adding, so essentially the same circuits are used. These circuits form the basis of the following IC's:

7482 2 bit full adder
7483 4 bit full adder
74181 Arithmetic Logic Unit

The 74181 is a complete Arithmetic Logic Unit with two 4-bit data buses and 4 control lines which can do addition, subtraction, bit shifts, comparisons and a number of other things. It is important to realize, however that for all its complexity, the outputs are at all times uniquely determined by the present state of its inputs. If you cut one open you will not find inside a tiny Japanese Gentleman doing sums on a calculator – all you will find is NAND gates and inverters! If you did find a Japanese Gentleman sitting there, I think you would find the NAND gates inside the Calculator!

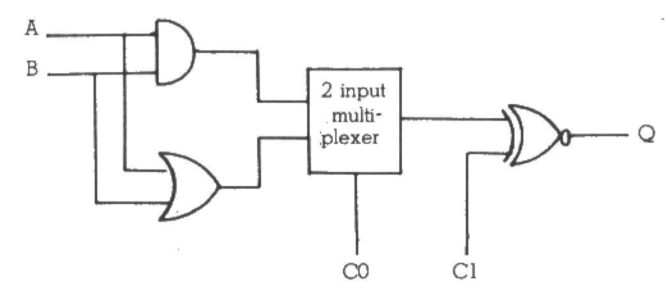


Fig. 9 The general purpose gate (1)

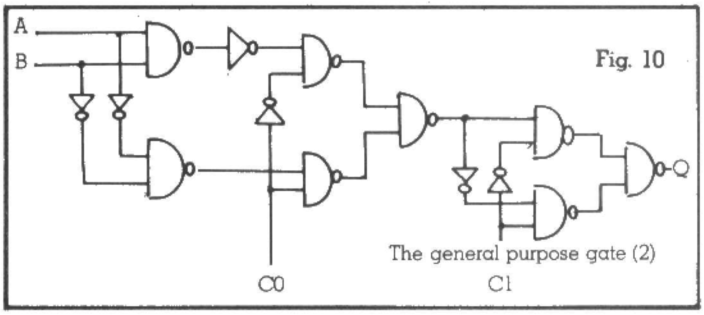


Fig. 10 The general purpose gate (2)

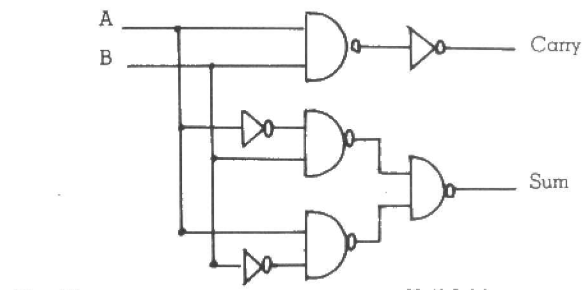


Fig. 11 Half Adder

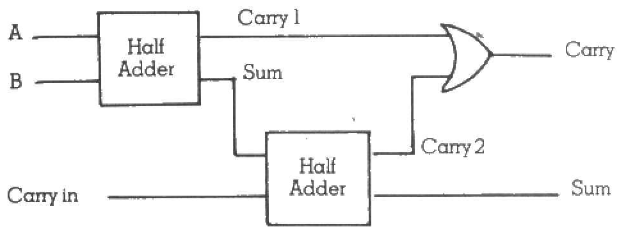


Fig. 12 Full adder

ERRATUM

The equations in last month's article were incorrectly printed and should be as follows:-

$$Q = \overline{A}.(B + C)$$

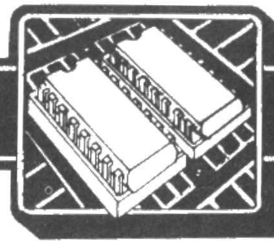
$$Q = \overline{A}.B + \overline{A}.C$$

$$A.B. = \overline{(\overline{A} + \overline{B})}$$

$$A + B = \overline{(\overline{A}.\overline{B})}$$

$$Q = \overline{\overline{A}.(\overline{B}.\overline{C})}$$

$$Q = \overline{A + \overline{(B + C)}}$$



TV To RGB Monitor Conversion

by Brian V. Alderwick, G8TCV

There are three normal ways of interfacing a computer and a television or visual display unit.

(1) To modulate the composite video information onto a UHF carrier to be fed into the aerial socket of a normal television.

(2) To send the composite video direct to a visual display unit.

(3) To send red, green, blue and sync signals separately to a monitor.

The first method is the easiest with most home computers, but carries the penalty of suffering from intermodulation effects causing patterning of the screen. Other disadvantages include non-saturated colours and unsharp display caused by reduced bandwidth inherent in the system.

The second method gives a great increase in the display quality but still suffers from some reduced bandwidth and, of course, needs a special video monitor.

The third method is by far the best, giving a sharp display with well saturated colours. The disadvantage is that a special monitor is needed which is expensive and useful only for that purpose.

This article describes an interface suitable to adapt a Ferguson 14" Movie Star portable television to an RGB monitor. All normal off-air functions are preserved and automatic switch-over to monitor occurs with switching on of the host computer.

The author's computer, a BBC Model B, has red, green, blue and sync outputs (R, G, B and S) available at TTL level, along with 5 volts and 0 volts. The interface described should be suitable for any other computer with these outputs available.

This television, along with most modern sets, has a live chassis and it is of paramount importance to isolate any input from the computer from it. Two methods are available. The first is to put an isolation transformer on the mains input to the television and earth the chassis. As an alternative, the inputs from the computer can be isolated with opto-coupled devices. The latter approach was adopted by the author, keeping the weight of the set to a minimum for normal portable use.

Two versions of the Ferguson TX9 chassis have been produced so far, the earliest chassis coded 1001 with the video processing I.C. μPC1365C not being suitable for conversion. Sets coded 1040 have the video processing I.C. TDA3560. This I.C. conveniently has data inputs for teletext available on a PCB plug. A second plug includes the intermediate frequency disable pin, leaving only two wires to be soldered to the board.

The circuit diagram is shown in Figure 1. The red, green, blue and sync inputs from the computer are fed through a buffer IC1. This has inverting open-collector outputs switching the opto-isolator diodes. The opto-isolators IC2-5 are Hewlett-Packard 6N137, high speed devices capable of working up to 10M Bits/s ensuring excellent bandwidth for the display. Although these devices are not cheap, the temptation to use lesser alternatives will result in a degraded display. The outputs from IC2-5 are fed into another open-collector inverter IC6. The R, G, and B outputs can be tapped off at a suitable level whilst the sync undergoes another inversion to make it compatable with the requirements of the television. The 5 volt supply needed by the television side of the interface is derived from a 12 volt supply through IC7, a 5 volt integrated regulator. The three relays, RL1-3, are switched by the computers 5 volt supply and switch the I.F. disable, sync output and data enable output, respectively.

Figure 2 shows the layout of the PCB and all components are mounted on it as shown in Figure 3. Construction is to standard practice and can include sockets for IC1-6. Do not forget the four wire links.

The input connections can be made by 6-way flat cable, enabling it to be passed through the aerial socket hole in the rear cover and, on the author's set, joined to a 6-way DIN socket. A cable can then be made up to go between this socket and the computer socket.

The output connections can also be made with flat multi-way cable to two sockets and two soldered connections.

The output connections are as follows:

SK15	Pin 1	0 volts
	Pin 2	BO Blue
	3	GO Green
	4	RO Red
	5	No Connection
	6	D Data Enable
SK6	8	I.F. Disable
	9	No Connection
	10	No Connection

Connect: 12v input to junction pad of L75 and R103. SO sync output to junction pad of C156 and R204.

These pad references are marked on the top side of the PCB in the television. SK15 fits an existing board plug marked PL15. Make sure that it is fitted the correct way round. SK6 fits on a board plug marked PL6, this being a ten-way plug, so make sure pin 8 is fitted to the correct pin on the board.

Initial setting up consists of pre-setting the variable potentiometers. Set RV1-3 one eighth travel, RV4 and RV5 two-thirds travel all in a clockwise direction. The board can be mounted on the left-hand side of the cabinet, looking in the back, with two nylon bolts or similar on a convenient moulded bracket.

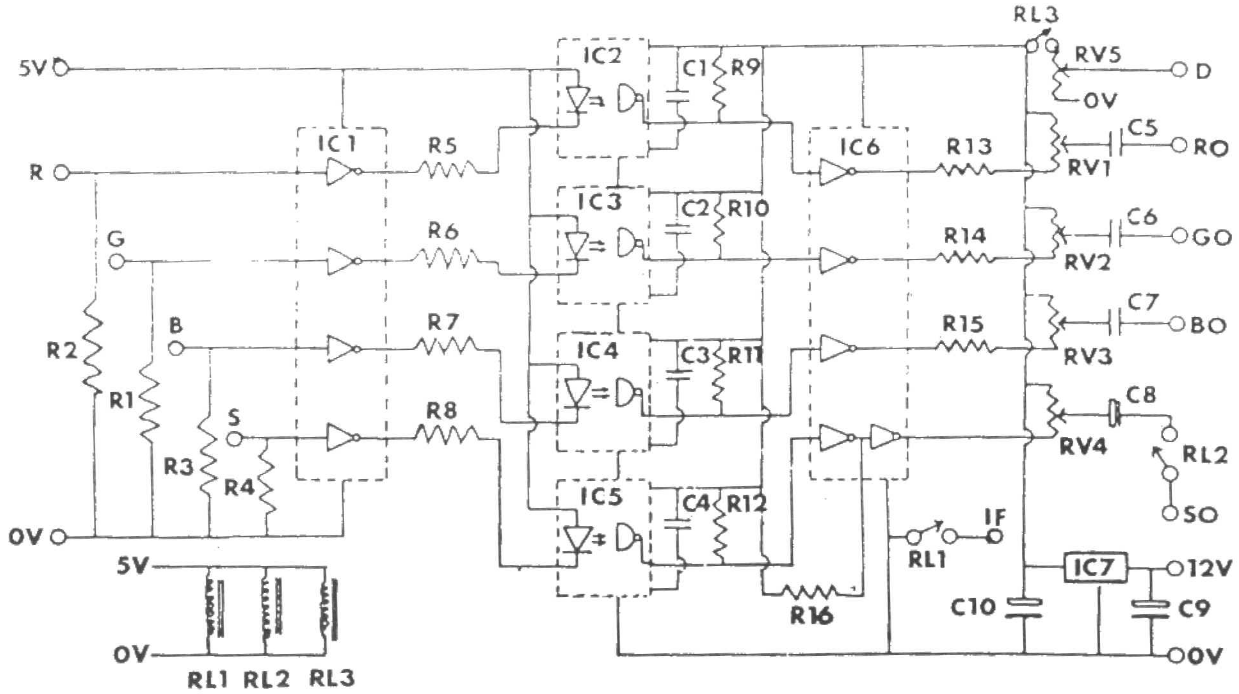


Figure 1
Circuit Diagram of Complete Interface

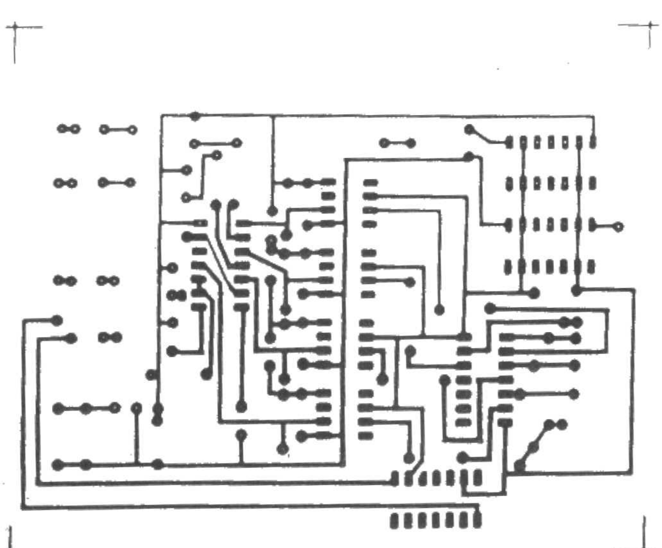


Figure 2
PCB Layout Foil Side

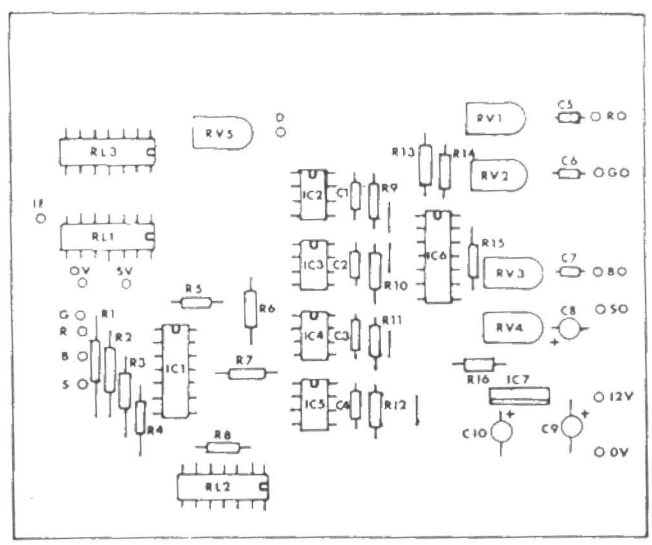


Figure 3
Layout of Components

With all the connections made, confirm the normal picture is there. When the computer is switched on, the screen should blank out and the normal picture be replaced by computer message. If the display is rolling, adjust RV4 until frame locks. At this stage the R, G, B levels, set by RV1-3, will need adjusting to obtain correct level and colour balance. To aid in this, a program for the BBC computer is given in Figure 5 which produces colour bars. Note – these adjustments are easy, but extreme care should be taken because of the live chassis and high voltages found in a TV set. Use an isolated tool to adjust these potentiometers – you have been warned!

Figure 4 gives a complete list of the parts necessary to build the interface.

In conclusion, in the author's opinion, a side-by-side comparison with a 14" colour monitor produced indistinguishable results. The total cost of the television and interface is less than the cost of a monitor and one also has a portable television as a bonus.

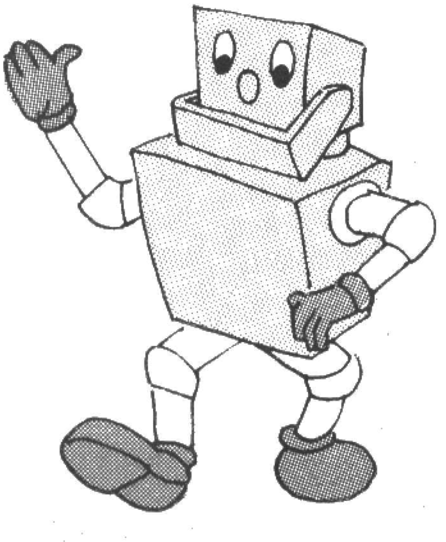
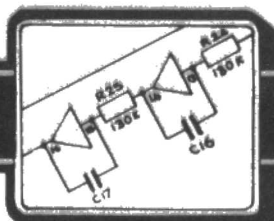


Figure 4 Parts List

R1-4, R9-12, R16	1K0 9-off All Resistors ¼ watt
R5-8	560R 4-off
R13-15	100R 3-off
RV1-3	100R 3-off
RV4	5K0 1-off
RV5	1K0 1-off
C1-4	.01μF 4-off
C5-7	100nF 3-off
C8	10μF 10v 1-off
C9	22μF 16v 1-off
C10	1μF 10v 1-off
IC1, IC6	7405 2-off
IC2-5	6N137 (or R.S. 304-273) 4-off
IC7	7805 1-off
RL1-3	R.S. 349-383 3-off
P.C.B.	3¾" x 4½" Single Sided 1-off
6-way Socket	Wafercon (Maplin HL11M) 1-off
3-way Socket	Wafercon (Maplin HL09K) 1-off
Terminals	Wafercon (Maplin HL14Q) 9-off
14-Pin D.I.L. Socket	2-off
8-Pin D.I.L. Socket	4-off
Wire for Links	

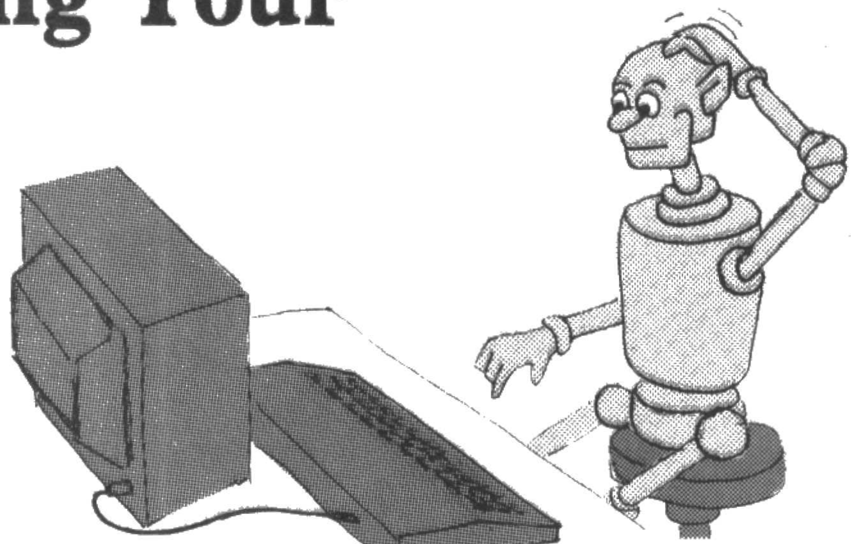
Figure 5 Program For Colour Bar Chart

```
 1Ø REM COLOUR BAR CHART
 2Ø MODE 2
 3Ø FOR I=Ø TO 112Ø STEP16Ø
 4Ø PROCfill (I,I+159,I/16Ø)
 5Ø NEXT I
 6Ø END
 7Ø DEF PROCfill (X1,X2,C)
 8Ø GCOL Ø,C
 9Ø MOVE X1,Ø
1ØØ PLOT5, X1,1Ø23
11Ø PLOT 85,X2,Ø
12Ø PLOT 85,X2,1Ø23
13Ø ENDPROC
```



Understanding Your Computer

by D. Boyde-Shaw



DISPLAYING INFORMATION

VIDEO DISPLAY

Many micro-computer users, especially in the home, use a TV set as the display device. Too small a screen size will, I think, cause a certain amount of eye strain and consequent headache. Too large a screen size and the display is overpowering, when you consider that you sit between two and three feet from it. The ideal size of screen would, I suggest, be 14 inch (36 cm), and don't forget screen size is measured in inches or cm, across the diagonal of the tube screen and is not necessarily the illuminated picture area, which could be up to half an inch less.

Part of the computer's memory is given over to screen information and is called video memory.

The display area of the TV screen is divided into a given number of small areas called 'pixels', which join up together to make a character area. They are usually laid out like a sheet of graph paper, although of course the graph paper is never seen on the screen.

There are many different combinations of pixels as regards the number in a row and in a column for example:-

The Texas 99/4A has 256 in a row and 192 in a column, and 64 pixels go to make up a character area, that is an 8 pixel by 8 pixel square.

The ATARI has 320 x 192 and the BBC 256 x 192.

Usually you can instruct the computer to switch each pixel on or off and so create a given shape within each character area, see fig. 1. The ZX81 however has only a pixel array of 64 x 44, and so resolution, as it is called, is low on the ZX81 but high on the Texas, Atari and BBC.

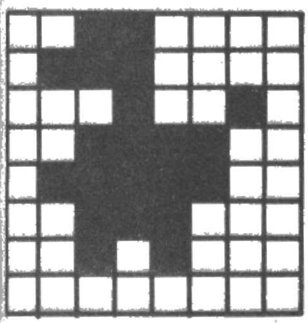


Fig. 1

Texas character area — black squares switched off —

To complicate matters further, in the more expensive micros, the on and off state can each be coloured.

For example on the Texas each character (64 pixels) can have one of 16 colours – and black, white and transparent are usually called colours, so in theory you can have 16 times 16 colour combinations – that is 256 different arrangements.

In addition to this the screen can also be coloured in the same 16 colours and so leads us to assume that we can have 16 x 256 combinations – 4096 different arrangements – quite a thought. But remember that a number of combinations could lead to a blank screen no matter what you wanted displayed.

For example in Figure 1 above if ON was green and OFF was also green and the screen also green, you only end up with a green screen. Also if either ON or OFF were transparent in the previous colour arrangement. It is a fascinating study but too deep to go into any further here. So every time you press a key on the input device or a programme runs, each video memory space will have a character coded for it and this character will appear on the screen, either in the 'default' colour or the colour combination you have programmed. Default means the colour combination, the ROM, has been instructed to tell the screen to display information in, often one colour for listing the programme and one for running it.

All this of course, leads to the modern micro being capable of high resolution graphics with full colour, to be discussed in greater detail later.

Printing Display

Naturally the video screen is a temporary print out and can be removed at a moments notice by the necessary input information. For a more permanent display or record the contents of the screen can be printed on to a sheet of paper by a printer. The contents of RAM either in whole or part can also be sent to the printer for printing – it all depends on your input information. The printer produces what is called 'hard copy' and the video screen 'soft copy'. There are 3 main types of printer: the daisy wheel, the thermal matrix and the dot matrix. All of them work by being informed by the computer which character they should print at any time, but they have varying degrees of noisiness and quietness and ability to print a variety of characters.

The thermal is the quietest and the daisy wheel has limitations regarding character printing, if you invent an unusual character the daisy wheel couldn't print it, because the character could not be on its wheel, similar to a typewriter problem. The other two have no such limitations.

The dot or thermal matrix printer receives information from the computer telling its character head which dot in the matrix to 'switch on' and which to 'switch off'. On the dot matrix the 'pixels' push forward for on and stay back for off and on the thermal they are either electrically on or off so as to heat up the special paper used to print the required character. Dot matrix types on 'normal' paper.

So we have seen that video displays are temporary and the printer naturally produces a permanent one. With the exception of the ROM the printer is the most permanent part of a micro computer system. The next article deals with computer languages.

Languages

As touched on in the first article the computer uses what is called a machine language – the computers own foreign language – the language one computer would use to talk to another computer.

It would take a long time and be pretty boring for humans to talk with computers in machine language, though of course it is done and examples of it can be seen every month in magazines such as this. We, the uninformed, therefore use a language similar to our own to tell the computer what to do, this high level language, as it is called, is itself called BASIC. BASIC stands for **B**eginners **A**llpurpose **S**ymbolic **I**nstruction **C**ode, and was designed by Thomas Kurtz and John Kemery at Dartmouth College in the United States during 1963/64 to introduce students to computer programming. Since then its use has spread world wide and is now the most popular introdubtory programming language used, in industry, business and education. Basic was designed to produce a simple interchange between man and the computer and the recent graphic capabilities have enhanced this ability even further.

But there are many versions of Basic, many dialects on a basic theme, so in 1978 a European and American Standard was produced to try to standardise the language, but to no avail.

Computer manufacturers always try to outdo other computer manufacturers in the versatility of their Basic and we, the users, are still left today with too many versions on the market; and some of the original ideas of simplicity have been lost. A programme written in Basic is made up of a set of instructions called statements, each one preceded by a line number, which tells the computer in what order to execute the instructions. The example given in an earlier article, reproduced below shows this:-

```
 5  PRINT   "WHAT IS YOUR NAME?"
10  INPUT   X$
15  PRINT
20  PRINT   "HELLO "; X$ ; "."
25  PRINT
30  PRINT   "TODAYS DATE IS"
35  PRINT   20th FEBRUARY 1983
40  END
```

Each line consists of a line number, usually in steps of 5 or 10 to allow for additions when editing, followed by an instruction word, which tells the computer what it has to do.

Basic has about 20 instruction words, and line numbers are only limited by the size of RAM available. 1K RAM should allow for 40 lines of programme – being 1024 bytes or characters.

So a programme of 40 lines using 1K RAM could hopefully have 1024 characters in total or 1024/40 per line, about 25. If some were shorter, like lines 15 and 25, others could be longer, and so on. Example instruction words used are:-

PRINT	– tells the computer to DISPLAY on the screen what is in the instruction.
READ	– tells the computer to read some DATA
DATA	– gives the computer certain values which it must use.
REM	– tells the computer not to use this information it is a REMARK for the programmers memory.
INPUT	– tells the computer to ask the outside world for some information via the keyboard.
LET	– assigns a value to a variable.
GO TO	– tells the computer to go to a given line number.
GO SUB	– tells the computer to go to a special sub programme named by a line number.
IF-THEN-ELSE	– tells the computer to carry out the following: IF – a certain condition – THEN – go to a given line number – ELSE – go to another line number.
END	– tells the computer to wait for the next run of the programme.

GO SUB is usually associated with a **RETURN** statement at the end of the sub programme to return the computer to the main programme.

DATA is in two forms – a numeric value or a string arrangement, and is either a constant or a variable.

String constants are enclosed in quotation marks (see lines 5, 20, 30, 35 in fig. 1) or

"I AM A STRING CONSTANT"
" 0272 851337"
" " (a null string)

String variables require the dollar sign $ on the end (see line 10, 20) or A$ P7$

If the value of the data is specific a constant is used for example:-

2, –3, 2.784, –21.7

If not its a numberic variable and is like a symbol in algebra for example: B, P3

So basically that is BASIC, a reasonably straightforward method of communicating with the micro computer.

There are in existence a goodly number of other languages, both specialised and general. For example:

FORTH
LOGO
PASCAL
COBOL
FORTRAN
ASSEMBLY
LISP

Fortran, Cobol and Pascal are very powerful structured computer languages available on a variety of micro and mini computers. Pascal is available on Texas Apple and the Atom, Fortran and Cobol on Video Genie and Apple, for example:

Logo is an educational language available on Texas and a few others.

LISP is available on the Atom.

Forth like Basic an extremely versatile language but which you can adapt to your own requirements, unlike Basic.

Words given in ROM can be redefined to produce new words as required which can again in turn be used to produce further words, and so on. It is about 10 times faster than BASIC, which is slow, because all the instructions must first be interpreted by the ROM to produce machine language; a slow process. Forth rearranges the instructions of Basic to get the job done a whole lot quicker.

For example if in Basic you put in:

PRINT 6 + 8,

the computer would translate this first into machine language and then put the 6 and the 8 into the ALU and find the answer by arithmetical means and print out the answer, after retranslation, of 14.

In Forth you only type in 6 8 +. , which tells the computer to put in 6 then 8 and add together. The result is left on top of the memory stack as 14 and is printed out.

At the moment at the cheaper end of the market the Jupiter ACE uses FORTH, which stands for Fourth Generation Computer language. Forth is easily learnt and was adapted by Richard Altwasser and Steven Vickers who produced the ACE from the language FORTH 79.

Assembly language is a bit of a hybrid. It uses a code instead of a language which is translated into machine language by a compiler, not an interpreter. It makes for quicker input and action on the part of the computer.

Basic is the most convenient for the most of us, but it is the slowest, when all is said and done.

Next time we'll talk about colour and graphics.

Articles Four to Eight

FOUR – Colour and Graphics

Graphics – How the computer carries out graphics commands. Use of gracs in a programme, how graphics can provide visual explanations and directions.

Colour – How colour enhances graphic and written programmes – highlights – use of grey, black and white.

FIVE – Speech and Music

Speech – Use of speech in educational programmes, making games more effective.

Music – Writing music – making sounds – how the computer makes sounds and music.

SIX – Simple games

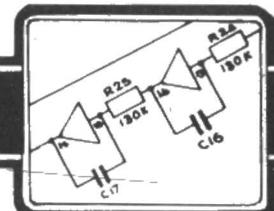
– The idea behind the games programme, using graphics and colour. Using speech and music and noise. Using the joystick and/or direction control from keyboard. Types of games.

SEVEN – Storage

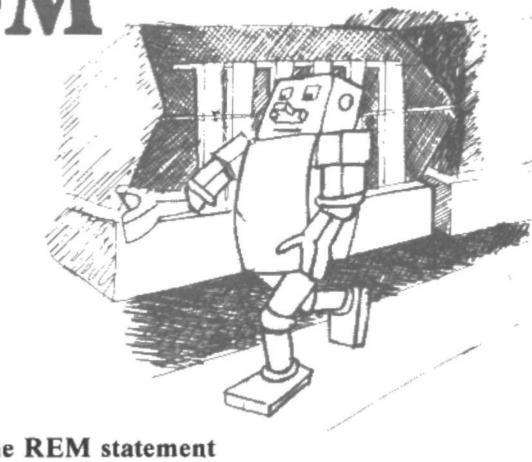
– Cassette
the hows and whys
Disk

EIGHT – Tips and winks

– Helpful hints on programming. Glossary of terms used in computing. Description of common micros.
Guide to the best buy.
Uses of the micro in the home and education.



INSIDE THE ATOM



A tour inside the machine by the men from PROCYON

PART 1 – Pirating and Protection

by J. R. Stevenson and J. C. Rockett

Most ATOM owners are as much 'hobbyists' as they are computer users, and ACORN have now recognised this. Many owners are keen to learn how the machine operates at levels other than through BASIC, and this series is meant to help you do just that. This is the first in a series of articles taken from the ACORN approved book "Splitting the ATOM", written by us. We hope this series will make the machine 'transparent' to you, in terms both of its hardware and its firmware (ROMS). We start with two articles on program protection. Future articles will include: your own memory display/modify program; how the interpreter works; how the stacks work; working examples from ROM routines; writing your own BASIC words; the structure of tape files; building your own ROM switch, and many more. But let's get to it straight away.

Programs saved on tape by ATOM can be protected from copying by several general methods: i) Prevent the program from being LISTed; ii) when the program runs it alters some part of the machine's cassette operating system (COS); iii) load the main program by using a 'preloader' program involving some machine code. Several techniques for each of these methods are given.

Of course, there is no way to totally protect a program other than by using a mathematical 'trap door' function, and these are unsuitable for small machines. Thus any program can be copied if a pirate has the right hardware and software, plus the skill, for the job. As with most things this is a two-edged sword, since the techniques for preventing copying are the same as for pirating.

In general, a machine-oriented chip such as DISATOM will allow any of the protection techniques given to be overcome by a skilled user. If you do not have one in your machine, use the HEX DUMP program in Part 2 of this series, which allows you to directly inspect and modify memory. Although this is more awkward than using a ROM, it's better than no tools at all.

The examples and techniques below are in order of increasing complexity. In all cases the following symbols will be used, and virtually ALL NUMBERS WILL BE HEX!!

<....> = push this key, such as <space> or <CR> or <CTRL-C>.

[XX] = an actual byte in memory, as a HEX number. e.g. [0D] or [03 CD 9F].

A) Using the REM statement

1) Start the program with:

10REM<CTRL-L> <CTRL-C> <CTRL-U> <CR>

This clears the screen, turns off the printer, and turns off the VDU screen. As you type <CTRL-U> the screen is disabled, but carry on typing the line, then type <CTRL-F> <ESC> and the screen is re-enabled. Now any attempt to LIST will send the control characters behind the REM to the print stream, and they will take effect. However, a RUN is still OK, since BASIC disregards anything after the REM. It is easier to insert the control codes directly into memory using the DISATOM, but this should all be done after the program is completely perfected. For example, using DISATOM type in

10REM<space> <space> <space> CAN'T <space> <CR>

Then type A 2900<CR> <REPT> <ESC> .

This will give an ASCII DUMP of memory at 2900 as follows:

A	2900	0D	00	0A	.R	.E	.M	20	20
A	2908	20	.C	.A	.N	.'	.T	20	0D

Move the cursor up to the A 2900 and <copy> the line over to the first 20. Change this and the next 20 to 0C 03, then hit <CR> and <ESC>. In the same way edit the final 20 (before the 0D) to 15 (see the appendix on the DISATOM toolkit for further details on its use). After editing, an ASCII DUMP gives:

A	2900	0D	00	0A	.R	.E	.M	0C	03
A	2908	20	.C	.A	.N	.'	.T	15	0D

Now any attempt to LIST will clear the screen, the word CAN't will appear in the upper left corner, and the printer and screen will be turned off.

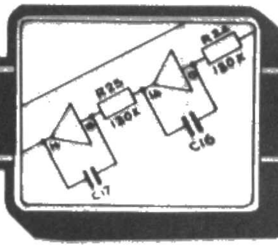
Unfortunately, since this must be physically the first line in the program, a pirate can overcome it by simply typing

0 <CR> L. <CR>

and if this has no effect he recovers the screen with

<CTRL-F> <CR> followed by
1 <CR> L. <CR> and so on.

This has the effect of eventually removing the BASIC line with the offending REM statement in it. Alternatively, if the pirate has DISATOM, he may do an ASCII DUMP and replace the 0C, 03 and 15 with 20's.



2) A REM statement can be used after a genuine BASIC statement. The REM is followed by four deleting backspaces [08], and then some apparently legitimate BASIC statement such as X=3*A, then a [15] (screen off). The line is best set up by typing out

```
10DIM XX(12);REM <3 spaces> X=3*A <space>
<CR>
```

The first three spaces [20] are then replaced by [08] (backspace), and the final [20] by [15] (screen off). When a list is attempted, the following appears on the screen:

```
10DIM XX(12);X=3*A (screen fails)
```

The dimensioning of the array is genuine, but the second statement is a fake. The purpose of all this is to convince the user that the entire line is real, and leave him baffled as to why the screen failed. It can of course by overcome by an ASCII dump, which would reveal the REM, and it can then be removed. However, if someone attempts to delete the entire line (as in the last example) the entire program fails. You can see that there are several possible twists in this technique.

A slight sophistication is to have another REM as the last line of the program that reads

```
10000REM[06]
```

Now when a LIST is called this results in

```
>LIST
10DIM XX(12);X=3*A
```

and the program appears to have only one line. The technique can again be defeated by an ASCII or HEX dump that reveals either the first or last REM.

3) Machine Code in a REM statement

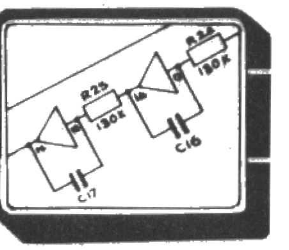
Quite a lot of machine code can be put in a disguised REM statement. As with the previous example, the first part of the line is valid BASIC, but buried in the REM is some machine code to be accessed by a later part of the program. Thus

```
40X=6;?18=41;REM[08 08 08 15 < m/c code here >
06 0D ]
!! M/C CODE MUST NOT CONTAIN 0D!!
```

The first two instructions are valid and appear on LISTing. The REM causes three backspacing deletes and turns off the screen so your machine code is not seen, then turns it on again at the end. The line can be placed anywhere in the program, but the deeper in the better, since this decreases the likelyhood that someone will stumble on it with an ASCII dump. The m/c code can be anything at all, but for example, it might alter the SAVVEC of the COS system to disable the tape saving function. There are two disadvantages to this method: i. you must exactly determine the entry point for the hidden machine code (then set P equal to that address and have it assembled there), and ii. you must eventually LINK to that address. Someone seeing a LINK into the BASIC text area will of course be suspicious, and in any event all LINKS can be found with a DISATOM using FIND"LINK" and FIND "LI.". It is possible to access this code via another, less suspicious m/c code routine. Using this method without the camouflage is a good way to save short machine code routines within BASIC itself, instead of having to assemble it each time, or using *SAVE to ensure machine code outside BASIC text is also saved. To prevent someone hitting <break> and then copying, site the BASIC program to start at 2800,, and then having a hidden REM that contains NOP;RTS([EA 60]) such that the NOP is at 2900 and the RTS at 2901. On one occasion in the program proper, LINK to 2900. If a pirate breaks from the program and then copies it, the 2900,2901 machine code is lost, and the program will crash. An even more effective way is to have 28FF=JMP 28XX, and somewhere in the 2800's is another REM containing an RTS. Hitting <break> distorts the JMP location and the program crashes.

Indirect jumps can also be used, via an address stored here. Once the program is running, it is easy to prevent the <ESC> key operating by intercepting the code from the keyboard and changing it. This is done by:

```
LDA@0 ; STA#B000 ENABLE the keyboard
JSR#FE94          GET a key in accumulator
CMP@#D            IF a <CR> jump to DISABLE
BEQ P+8
CMP@ 32
BCS P+4           IF >= <space> jump to
                  DISABLE
LDA@ 32           CHANGE code to a <space>
PHA
LDA@ 10;STA#B000DISABLE keyboard again
PLA
RTS
```



Finally, spanner the vector at 20A, 20B to point at this routine. The routine also prevents entry of any other control codes, some of which re-enable the <ESC> key. Remember to set B000=10 as the first part of your basic program. This can be beaten by causing an error, which will return the user to the direct mode. To be safe you should therefore alter the BRK vector at 202,203.

4) The Long Line

The BASIC interpreter is perfectly happy to work on a line which is (almost) infinitely long, with the statements being separated by semi-colons. The practical consequences of this are that i) the LIST command will turn back on itself (recycle from the start) if the line is greater than 258 bytes (two of these are the line number), and ii) If this is the first line in the program then BASIC is unable to add any new lines or delete any old ones, since it cannot find the end of the first line. If the first line consists of something like P. ;P. ;P. ; etc. etc. for the whole of page 29, then the rest of the program cannot be LISTed and the program cannot be edited, nor do the commands OLD or END work, since the real size of the program is now unknown to the operating system. The real program can be terminated with LINK#C2B2, which accomplishes a NEW, or a GOTO X, where X is a real line number or label in the program, if you wish to repeat the program. Below is a procedure for setting up such a method, and it is given so that those without DISATOM can also do it, given some extra work. Make sure that your program is perfect BEFORE you protect it, and note that you have 258 bytes less space for your real program.

1) DIRECT COMMAND:
```
F.I=#2900TO#2A04S.4; ! I=#3B202E50;N.
```
2) DIRECT COMMAND:
```
?18=#2A
NEW
```

Now write and completely debug your program as normal, but the first line must be 1 REM <3 spaces> <CR> . <CR> .

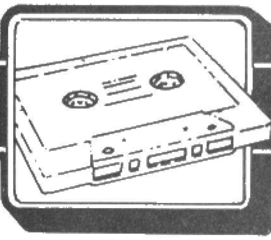
3) When your program is perfect give DIRECT COMMANDS:

```
?18=#29
!#2900=5000000D
!#2904=#3B20202E
!#2A00=#3B202E50
!#2A04=#20202E50
```

*SAVE the program in the usual way, remembering that the total program does start at 2900.

Next week we'll be telling you about more advanced techniques such as pre-loaders, and we'll be giving you a memory dump/modify program to put all this into practice!!

The book 'Splitting the ATOM and the chip DISATOM are available from PROCYON, 57 Westgate, Cleckheaton, W. Yorks. Telephone: (0274) 588192).



BBC SOFTWARE

Reviewed by S. M. Gee and Kay Ewbank

The BBC Micro has been around for just over a year and in that time it has been shown to be a really super machine that has capabilities that go way beyond those of most other home computers. As yet, however, the BBC Micro has not really made its full impact. In part this is due to initial problems of delivering enough machines to satisfy demand, but it is also because it takes time to develop software of the quality needed to show off all the programming techniques that are available.

The BBC Micro is a fast machine – even in BASIC – and the way it allows the programmer to mix BASIC and machine code is also a great advantage. The fact that the BBC Micro is such a good machine does mean, however, that the programmer never has any excuses for leaving programs in an unfinished, unperfected state. There comes a point, when programming on any lesser micro where you have to give up adding those extra routines that make programs pleasant to use on the grounds that they would cause the program to overflow or to run too slowly. But you can carry on improving your BBC Micro programs a lot further without any such reservations. So the standards set for BBC Micro software are therefore more exacting.

After an early dearth of software, there is now a veritable flood of it and out main problem in reviewing it was simply how to select from among the more than a hundred cassettes that arrived. In some ways it was easier to decide which programs to reserve for more extensive consideration in future issues of E & CM. Beebugsoft's Utilities pack, Micro Aid's Payroll program, Acornsoft's Philosopher's Quest and Computer Concept's "Wordwise" (a word processor relying on a plug in ROM) all fell into this category on the grounds that they needed a long period for familiarisation. Others, like Clare's Beebsoft Joystik Pack will have to wait for us to acquire the extra hardware they require. Having put a few programs, aside though we still had many more tapes than could be covered even in an extended feature. In the end it seemed best not to make a choice as such for this first review feature, but just to pick some titles at random to try to reflect the range of products available. Over the coming months we will however scrutinise lots more of the many games, educational programs and applications packages so if the particular program you want information about is not mentioned here please look out for future reviews.

Almost every game you could ever imagine has now been produced for the BBC Micro – many of the popular ones in lots of versions – and the list of available software is growing daily. We were even sent some software in pre-publication form. It's always nice to have the privilege of testing out programs before anyone else has seen them so we took a quick look at six games currently being developed by A & F Software. These included a version of Frogger (not one of the fastest we've seen, though) and a novel and exciting variation on the invaders theme called Planes. There were also two adventure games that seemed promising – Pharoah's Tomb and Tower of Alos – a version of Lunar Lander that made good use of graphics and a Painter. We would have liked to have previewed Romik's first BBC Micro tape, "Birds of Prey" but unfortunately the pre-publication copy failed to load.

Loading cassette software can sometimes prove a tiresome problem but the BBC Micro is generally very reliable. We were able to load every commercially available tape supplied – most of them first time round. There was just one minor problem that as yet will only affect a minority of users but is worth bearing in mind for the future. We were using a Version 1.0 Model B machine with a disk drive whereas many of the software suppliers have developed their programs using cassette storage running under earlier versions of the MOS. The loading instructions, therefore, do not alwyas take account of the commands needed to set the machine up. For example, some programs need PAGE=&0E00 before loading to make enough RAM space available.

MONSTERS (Model B) and DESK DIARY (Model A & B) (??.??) from ACORNSOFT

It is already a well recognised fact that Acornsoft have set the standard of excellence for software written for the BBC Micro so in some ways you may find it strange that we start with a review of two of their programs. We did, however, make the decision to use Acornsoft as our 'benchmark' and then see how other software measured up to it. Monsters, like all the other Acornsoft games that we have seen, makes excellent use of the BBC Micro's colour, graphics and sound. The screen display is of walls and ladders with a little man figure and bug-like creatures. The object of the game is to trap the monsters by digging holes for them to fall into and then filling in the holes before they can crawl out. If it wasn't for the fact that we've seen other people play it successfully we'd probably declare it was fun but impossible. As it is, we'll just have to hope that practice makes perfect and perhaps we will eventually trap at least one before being eaten! Certainly, it lives up to its reputation of being highly addictive – a quality that it possesses by virtue of having fascinating graphics and exactly appropriate sound effects. All-in-all it is a top quality software product.

There are two programs on the Desk Diary Tape, an address book and a desk diary planner. We tried out the second one of these to discover how user friendly it was. It was soon apparent that it was very easy to use, that the display was well laid out on the screen with good use of contrasting colours, and that the program did not crash even when provoked! Both programs run on both Model A and Model B since they use Mode 7 – which is the appropriate choice for these applications. Although this is not a piece of software that we personally are going to use – preferring old fashioned methods for the time being at least – we were impressed by its overall presentation.

SWAMP MONSTERS (Model B) (£6.50 inc VAT and p & p) and INVADERS (32K) (£6.50 inc VAT and p & p) from MP SOFTWARE AND SERVICES, 165 Spital Road, Bromborough, Merseyside L62 2AE

There are lots and lots of versions of the most popular arcade games and of all the firm facourites Invaders has probably seen the most copies. The version from MP turned out to be easy and had little to recommend it. The graphics were effective but the sound effects were annoying rather than enhancing. We found it slow to respond to keypresses and did not like the eay the missiles fired 'off-centre'. When you lose a base you get a new field of invaders – which is annoying if you've almost managed to blast the old lot out of the sky. We crashed the program very easily and found ourselves left in fast repeating key mode and experienced another bug which meant that the game restarted before it was properly finished.

The fact that MP's program with the title Swamp turned out to have features in common with Acornsoft's Monsters was more of a surprise. Although nowhere near as interesting as its counterpart in terms of its graphics and sound, it did present some advantages to the novice at the game in that it was much slower moving and therefore much easier to play. In this game the player controls a robot who has to cross a swamp via stepping stones which he can pick up or replace in order to trap the monsters. There are far fewer options and the pace of the game is relatively relaxed. Even under these conditions, however, we did not survive for long!

BEEBSOFT GRAFKEY (32K) (£5.75 inc. VAT and p & p) from CLARE'S, 222 TOWNFIELDS ROAD, WINSFORD, CHESHIRE

The BBC Micro offers lots of possibilities for creating graphics displays and this piece of software has been designed to take advantage of them. It is not intended as just another etch-a-sketch program but is meant to have lots of useful applications for drawing charts and diagrams. It is a program which is easy to use to create impressive effects by simple straightforward steps. Although full instructions are displayed on the screen at the beginning of the program, it would also be helpful to have a paper copy of them for later reference.

GAMES OF LOGIC AND CUNNING (Models A & B) (£9.20 plus 50p p & p) from GOLEM, 77 QUALITAS, BRACKNELL, BERKSHIRE RG12 4QG

This tape contains five separate games but unfortunately we were not impressed by any of them. The first three were fairly standard, traditional games of strategy which, in terms of programming could be described as trivial. No attempt has been made to enliven these games by including graphics – which does, however, mean that they work on a Model A machine. We were unimpressed and unamused by the "amazing test of telepathy" apart from its more imaginative use of sound. The final game on the tape does use some graphics (and so only runs on the Model B) but did not rise much above the mediocre.

SUPERLOT (32K) from BEEBUGSOFT

Definitely a super program. It behaves nicely, responds quickly, produces clear displays, is amazingly clever and is really user friendly. What, you may be asking does it do? It plots graphs in three co-ordinate systems. If this sounds either boring or intimidating then think again – this package is just the sort of software that brings mathematics to life and should give computers a good name. All you have to do is specify the equation (or equations in polar co-ordinates for example), the number of points to be plotted and the range that x should vary over and Superplot does the rest. With its facility for overlaying one graph on another, it goes well beyond what is normally taught in schools and makes learning great fun. Written in BASIC, this is a nicely structured program that has good error handling. In other words, just the sort of programming that should be encouraged.

SNAKE (Model B) (£7.80 plus VAT) and CHESS (Model B) (£10.00 plus VAT) from COMPUTER CONCEPTS, 16 Wayside, Chipperfield, Herts, WD4 9JJ

This version of Snakes, a fairly traditional computer game in which you have to guide a snake and eat coloured dots to make the snake longer. What makes the game difficult is that as the snake gets longer it has a tendency to get in its own way! If you do happen to run the head of the snake into its tail then you are dead. To make matters worse there are square blocks that you must avoid, hitting one of them also kills the snake. If you really cannot avoid a killer block then you do have the option of blasting it out of existence but it will cost you points! Overall the game was fast and fun to play. It made good use of the BBC Micro's facilities and we particularly liked the way it played the "death march" when you failed.

Computer Concept's Chess was graphically pleasing and had features that make it worth playing against a computer – although you can also use it to play against another human and gain the same advantages. If you are trying to improve your game, probably the most useful options are the ones that allow you to take back moves and, better still, replay the game from the beginning – and play on from the point where you made your mistake!

CHESS (32K) (£6.95 plus 50p p & p plus VAT), WORLD GEOGRAPHY (Models A & B) (£5.95 plus 50p p & p plus VAT) and FOOTER (32K) (£6.95 plus 50p p & p plus VAT) from MICRO POWER, 8/8a Regent Street, Chapel Allerton, Leeds

We were also impressed by the Chess program from Micro Power. Although it actually presented fewer options (six levels of play rather than eight and only against the computer) the ones it offers seem to be adequate and it seemed easier to get started. As well as being able to

replay and analyse your own games you can also study a game between Spassky and Fischer to see how the World Champions play.

The good feature about the World Geography tape was the map of the world displayed in the top half of the screen. The program tests your knowledge of the capitals of the countries of the world and their populations and a dot appears on the map to indicate the location of the place in question. Two sound effects indicate a correct guess and a false one.

We had never played two-person football on a computer before and had great fun with Footer. High-res graphics are used for the display and the way the two players moved their arms as well as their legs was a nice touch. This was a game we wanted to carry on playing – which is presumably the acid test. However you have to remember that this is a two person game and so it is important to make sure that there is someone else willing to play it with you.

STATPACK (Model B) (£7.95) and FLAGS (£2.95) from MICRO AID 25 Fore Street, Praze, Camborne, Cornwall, TR14 0JX

Being familiar with the great IBM's "Statpack", we wondered just what a 32K statistics package would offer us. In terms of its facilities it is a fairly promising program but we found it not very user friendly and thought is presentation sloppy. The displays suffered from messy layout – the columns of figures did not line up – too many places beyond the decimal point – which is misleading as well as unnecessary since the computer cannot be accurate to seven decimal places – and a lack of graphics where they might have proved useful – in histograms, for example. The displays also scrolled up the screen, causing the user to miss vital information. The actual range of statistics options offered is also a little strange i.e. regression but not cross tabulation.

The Flags program was also poorly finished in that the text giving information about the country whose flag was being displayed sometimes appeared at one position, sometimes at another and was often half missing. It seems a shame to spoil a good idea by a lack of attention to this type of detail.

EDG GRAPHICS PACKAGE (Model B) (£24.95) from SALAMANDER SOFTWARE 17 Norfolk Road, Brighton, BN1 3AA

If you've seen John Cownie's book "Creative Graphics" (from Acornsoft) you'll know that the BBC Micro can produce amazing graphics displays. For many of us, however, even drawing simple shapes and pictures is a problem. This Graphics System package is therefore a potentially very welcome aid. It is a well documented piece of software that provides a short cut of exploring the grphics capabilities of the BBC Micro (Model B). The notes that accompany it give full information about how to use the program and they also include hints on how to produce perspective drawings. If you are already an expert computer graphics artist this package probably won't help you, but if you are not it will help you enjoy computer art and help you to create complex, multi-page pictures that will impress all who see them. Our only reservation about this software is its price. At nearly £25 it does seem expensive.

BRICK EM IN (32K) (£6.95) from SOFTWARE FOR ALL, 72 North Street, Romford

This is another well known computer game and we really enjoyed this version. It has a number of difficulty levels. We survived quite a long time at level 1 but were immediately overwhelmed by the attack launched at the highest level. The object of the game is to 'shunt' square bricks about the screen so as to trap the attackers. It sounds easy but the attackers move fast and one false move and they are on you. The display was colourful and clear, the sound effects enhanced the game well and there was a quick response to the keypress – a must for a fast moving game of this type. Its good micro entertainment for all the family – and must help improve your reaction times.

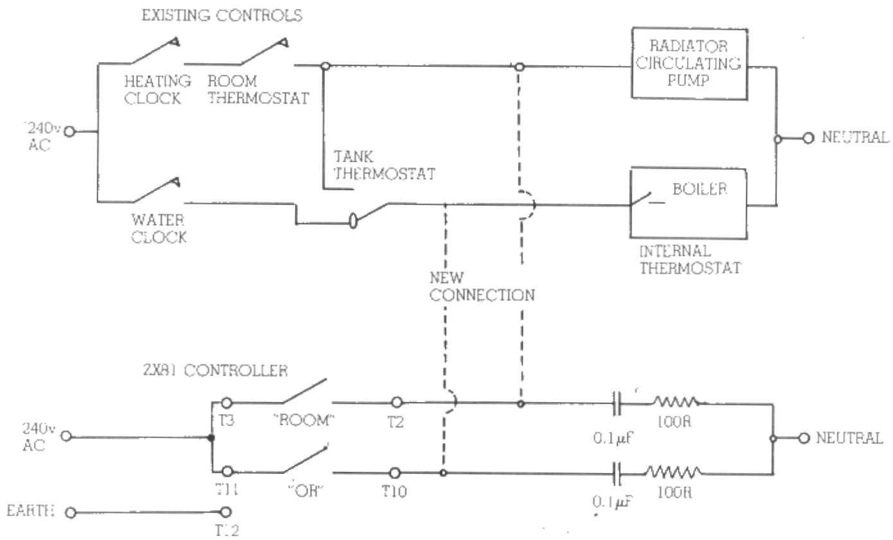
INKOSI (32K) (£5.95 inc.) and SEQUENCES (32K) (£5.95 inc.) from CHALKSOFT LOWMOOR COTTAGE, TONEDALE, WELLINGTON, SOMERSET

Chalksoft specialises in educational software for the age range 4 to 15. Inkosi is a fairly lavish implementation of the well known 'rule of kingdom for a few years' type of game. If you aren't familiar with this sort of thing then you haven't been getting full value from the Welcome pack which includes a simple but effective game called Yellow River Kingdom. There is no denying that Inkosi (the name of the ruling chief) is a very well written and implemented game of its type but we found it tedious because it gave the user very little to do. Sitting for minutes on end reading the screen fulls of annual reports of the state of the tribe before being asked to do anything is not a good way to become involved in what is going on. In our opinion Yellow River Kingdom is a better game – it is fast, it has good moving graphics, and it is supplied free with your BBC Micro – Inkosi is perhaps a better program but it fails as a game.

Sequences is the second offering from Chalksoft that we looked at and again we were impressed by the quality of the programming. However this time we found ourselves questioning the educational value of the program. By pressing numbers corresponding to a menu selection we were presented with a number of graphics frames illustrating how the most familiar number series are generated, for example squares, primes and so on. The graphics in general were good and there was plenty of sound. In fact so much sound that it became a little trying – if anything happened on the screen at least a beep would sting our ears. The main trouble with the program is that it presents no opportunity to interact with it. It could ask a few simple questions to see if you are following but it doesn't. This program is a direct translation of a blackboard type description of number series to the computer and as such is very good.

ZX Energy Management continued from Page 40.

Figure 1
A Typical Connection Scheme



A kit of parts including:

★ Printed Circuit Board

★ Plastic Box

★ Program Cassette

is available from:

ZX Energy Project,
E & CM,
40/42 Oxford Street,
DAVENTRY,
Northants.

Price: Full Kit (no separate instructions **£35**

Printed Circuit Board plus cassette without components **£10**

Figure 2
ZX Energy Circuit

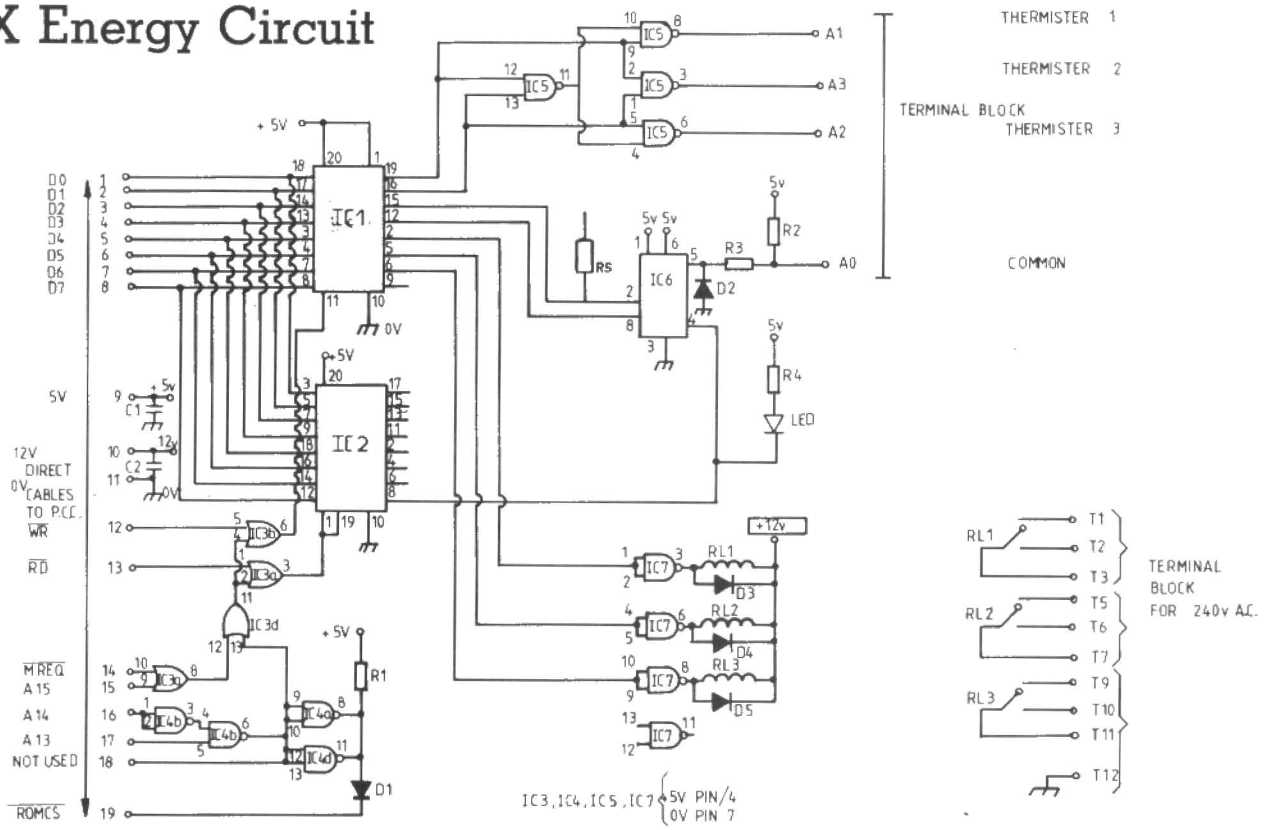


Figure 3
ZX Energy PEC Layout

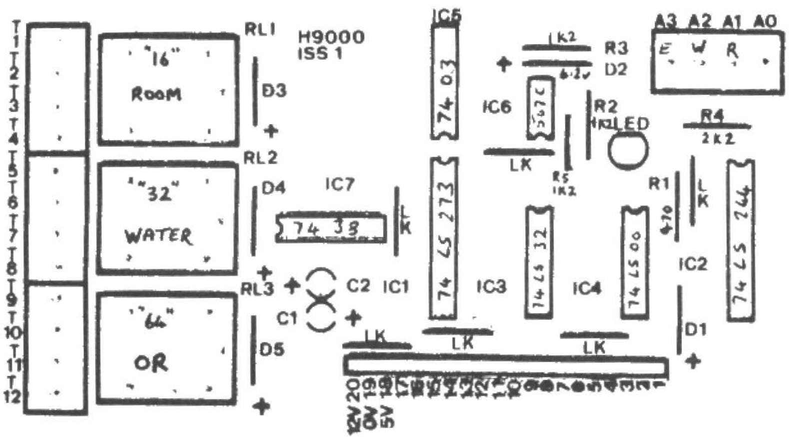
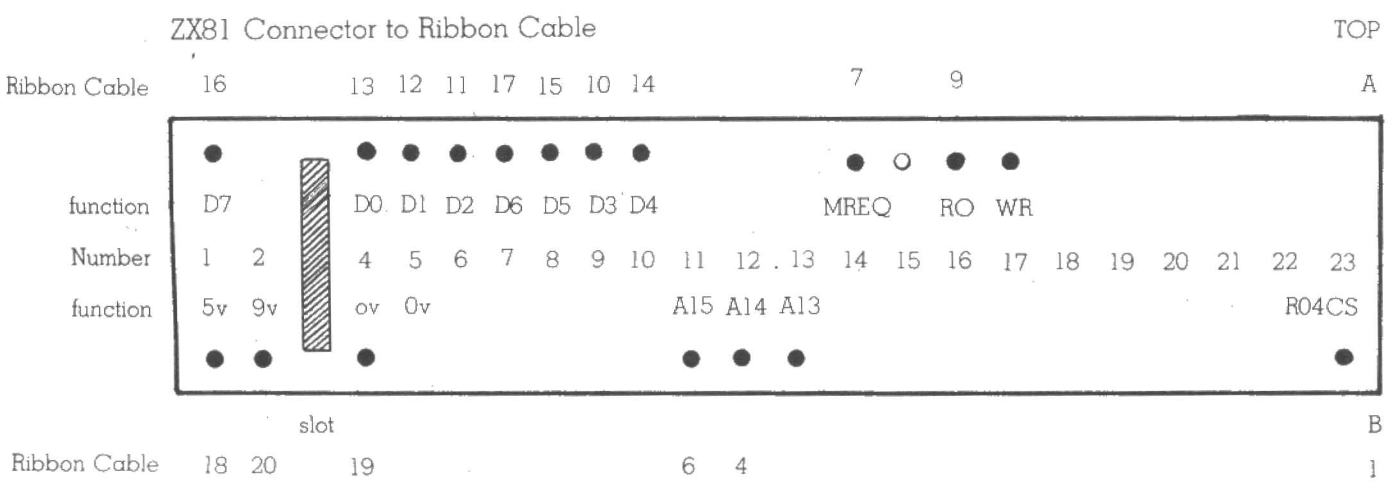


Figure 6
Program Variables

A Array of calendar data
B Water demand
C Clock (frames)
D Day number
D$ Days of week
E Extension time of Room demand (frames)
F Frames (50=1 sec)
G Variation for Water optimum start
H Hours
I Integer 1
J Max attainable room temp
K Port address (8194 to allow easy testing without ZXE)
L LSB of frames, also Room+DHW+Sensor bits output
M Minutes
O OR demand
P Room demand
Q loop counter
R Room set point
S "Stop" controlling
T Time (frames)
U MSB frames
V Variation for room optimum start
W Water set point
X temperatures array
Y Zone stop time (frames)
Z Zone number

Figure 4 ZX Energy



Notes: Ribbon cable 2 is not connected
Ribbon cable 5 is not connected

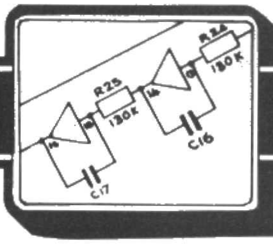


Figure 7 ZX Energy

```
STOP [illegible]      MONDAY 0:0

ROOM   -5.3 C       SET:0 C

WATER 31.9 C        SET:0 C

EXT    -5.3 C

ROOM EARLY          5 MINS

WATER EARLY         0 MINS
4306
[illegible]

FROM        ROOM  WATER
 0           0     0
 0           0     0
 0           0     0
 0           0     0
 0           0     0
 0           0     0
 0           0     0

  10 SLOW
  20 FOR N=I TO PI
  30 LET L=64*O+32*B+16*P+N
  40 POKE K,L+8
  50 FOR Q=I TO 127
  60 POKE K,L+4
  70 POKE K,L
  80 IF PEEK K>127 THEN NEXT Q
  90 LET X(N)=LN ((64+Q)/(192-Q)
)
 100 NEXT N
 110 LET X(I)=INT (160*X(I)+120)
/10
 120 LET X(I+I)=INT (260*X(I+I)+
500)/10
 130 LET X(PI)=INT (160*X(PI)+12
0)/10
1000 LET Y=432E4
1010 LET V=NOT I
1020 LET G=V
1030 IF Z>6 THEN GOTO 1500
1040 LET V=2E5
1050 LET Y= 18E4*INT A(D,Z+I,I) +3
E5*(A(D,Z+I,I) -INT (A(D,Z+I,I)))
1060 LET J=X(PI)+(70-X(PI))/INT
PI
1070 IF A(D,Z+I,I+I)<J AND X(I)<
J THEN LET V=8E4*LN ((J-X(I))/(J
-A(D,Z+I,I+I)))
1080 LET G=3E3*(A(D,Z+I,PI)-X(I+
I))
1500 LET R=A(D,Z+(T>Y-V),I+I)
1510 LET W=A(D,Z+(T>Y-G),PI)
1520 LET V=(V>I)*INT (V/3E3)
1530 LET G=(G>I)*INT (G/3E3)
1540 IF T>Y THEN LET Z=Z+I
2000 LET P=R>X(I)
2010 LET B=W>X(I+I)
2020 LET O=P OR B
2030 IF P THEN LET E=1E4+T
2040 LET P=E>T
3000 LET U=PEEK 16437
3010 LET L=PEEK 16436
3020 LET F=65535-256*U-L
4000 IF U>140 THEN GOTO 5E3
4010 IF L<10 THEN GOTO 3E3
4020 PAUSE I
4030 LET C=C+F
4040 LET F=I
5000 IF C+F<432E4 THEN GOTO 6E3
5010 LET C=C-432E4
5020 LET E=I
5030 LET Z=I
5040 LET D=D+I-D*(D>6)
6000 LET T=C+F
6010 LET H=INT (T/18E4)
6020 LET M=INT (T/3E3)-60*H
7000 PRINT AT NOT I,NOT I;"ZX EN
ERGY ";D$(6*D-5 TO 6*D);"DAY ";H
;":";M;" "
7010 PRINT
7020 PRINT "ROOM  ";X(I);" C   "
,CHR$ (CODE "█"*(P=I));"SET:";R;
" C "
7030 PRINT
7040 PRINT "WATER ";X(I+I);" C
",CHR$ (CODE "█"*(B=I));"SET:";
W;" C "
7050 PRINT
7060 PRINT "EXT    ";X(PI);" C
"
7070 PRINT
7080 PRINT "ROOM EARLY",V;" MINS
"
7090 PRINT
7100 PRINT "WATER EARLY",G;" MIN
S "
7110 PRINT PEEK 16404+256*PEEK 1
6405-16384
7120 PRINT "[illegible]"
7130 PRINT
7140 PRINT "FROM        ROOM  WATER
"
7150 FOR N=I TO 7
7160 PRINT CHR$ (CODE "█"*(N=Z))
;A(D,N,I);"       ";TAB 10;INT A(D,
N,I+I);"  ",INT A(D,N,PI);" "
7170 NEXT N
8000 IF INKEY$="A" THEN LET S=NO
T I
8010 IF S THEN GOTO I
8020 FAST
8030 PRINT AT NOT I,I;" STOP ";
8040 LET C=T
8050 LET E=I
8060 LET F=I
8070 PAUSE 4E4
8080 IF INKEY$="C" THEN LET S=I
8090 IF S THEN LET Z=I
8100 IF INKEY$="D" THEN LET D=D+
I-D*(D>6)
8110 IF INKEY$="Z" THEN LET Z=Z+
I-Z*(Z>6)
8120 IF INKEY$="F" THEN PRINT AT
 14+Z,I;"**"
8130 IF INKEY$="F" AND Z<>I THEN
 INPUT A(D,Z,I)
8140 LET A(D,Z,I)=ABS INT (A(D,Z
,I)*100)/100*(A(D,Z,I)<24)
8150 IF INKEY$="R" THEN PRINT AT
 14+Z,10;"**"
8160 IF INKEY$="R" THEN INPUT A(
D,Z,I+I)
8170 IF INKEY$="W" THEN PRINT AT
 14+Z,16;"**"
8180 IF INKEY$="W" THEN INPUT A(
D,Z,PI)
8190 IF INKEY$="T" THEN PRINT AT
 NOT I,20;"**.**"
8200 IF INKEY$="T" THEN INPUT C
8210 IF C<>T THEN LET C=ABS (18E
4*INT C+3E5*(C-INT C))*(C<24)
8220 IF INKEY$="S" THEN SAVE "█"
8230 GOTO 6E3
9000 LET K=8194
9010 LET I=SGN K
9020 LET B=NOT I
9030 LET C=I
9040 LET D=I
9050 LET E=I
9060 LET O=B
9070 LET P=B
9080 LET S=B
9090 LET T=I
9100 LET Z=I
9110 DIM X(PI)
9120 DIM A(7,7,PI)
9130 LET D$="     MON  TUESWEDNES
THURS    FRI SATUR   SUN"
9150 GOTO I
```

Continued from Page 45

Teddy Technicals Column

(From Deep in the Daventry Design Dungeon)

They locked us in there for the whole of last month, manacled to an MK14 (purgatory!), without a byte to eat!

We were busy brewing up acres of software, chipping away the silicon and frying them in midnight oil, while being lashed with rolled up back issues of Boys Own Paper – but now back again – here goes:-

Colin Rees of Dunstable asks can the graphics card be connected to a ZX81 and would it then be in colour. Yes and Yes in a future edition we will be dealing with connections to other systems. He says that he was boggled by the Tables in the Feb. issue. — so was I! Most of this is for reference only and it is quite easy to use – see next month.

Mr. Puddick asks if kits/boards are available because he does not want to use prototype boards – yes – talk to Stirling Microsystems (address in their ad.).

To run assembler you will need Ram card CPU card cassette card and tape based assembler and editor (also from Stirling's). Flex operating system includes an assembler and editor – Pascal is available for flex.

Mr. Lobb writes saying that our original cost estimates seem low compared with Stirling's prices.

We do not control Stirling's prices and neither do they write these articles! We suggest shopping around, for the components, would result in lower costs! On the other hand convenience must be worth something. The boards are available separately and the components are available from advertisers in this mag – pay your money and take your choice.

Mr. Cartwright of Colspie, Sutherland asks how to connect to a Dragon – well we will be covering this in a future edition but we would not expect to interface via Rom cartridge – thinking cap job this – watch this space. The card only looks like 20 addresses in Ram so it should be easy . . . ish! with a ribbon cable – stay tuned.

It would be possible to communicate to the Hi-Res computer via RS232 link (after all S-Bug and HT-Bug only supports RS232!). It would be possible to use the Dragon's cassette, drives, printer etc. if the data is transferred into the Hi-Res in S format (see Motorola manuals etc.). Use Dragon as intelligent terminal is another approach. I suspect that it would be easier to make the Hi-Res board share the Dragon's address space and to write or mimic your own routines. Copies of the source of the bugs will, probably, be available for purchase, later.

The 512 x 512 format should be able to be "bodged" to 256 x 256 by fitting a EF 9366 (which will fix format to 256 x 512) and by only fitting 4816's) to every other Ram socket and joining the Ram's o/p pin to the o/p pin of the next Ram socket (which has not been fitted). We end up with, effectively 16K x 4 per plance (because each bit is clocked out twice). The only draw backs with this are that only one page is possible (or two if 16K Rams are used but that would mean that ½ the Ram is wasted), the x register needs to be driven carefully because the value in x is twice that displayed (as is delta X). The Y registers are unaffected.

I haven't tried this, but it should work – no promises. If I do get this going one day I will publish it.

Some queries have been received regarding the operating system Flex-9. This is a well supported, very user friendly, powerful operating system which we consider to be much better suited to amateur systems than others we consider. Flex has nothing at all to do with the colour operation of the computer. It is involved with handling data, loading basic, contains an assembler etc., etc. It is analageous to CPM on other systems.

R. Daley from Lara, Victoria, Australia writes with many questions, some of which are answered elsewhere in the article. As he correctly says the cassette interface does not fit in the rack as it was so small (I bolted mine on one of the side plates). We have had quite a few letters on this and will track out the board to fit in the rack – OK? Watch this space!

Although we are using S-Bug (a South West Technical Products Product) not all SWTP products will work on the system, e.g. the Floppy Disc controller is at a different address and has slightly different hardware requirements. However the new system bugs are broadly SWTP compatable having the same I/O vectors etc. HI-Bug and HT-Bug support Flex, TSC Basic (the floppy boot progs are different from S-Bug). Thus SWTP and TSC's tape based products should run OK (ACIA is in the correct place etc.).

When boards are purchased from Stirling's design notes are available, suitably corrected and Typo-free.

We hope that this format is not too confusing and apologise not answering all the letters here but these are representative and many letter cover the same points.

How about some of you Antipodean readers forming a user group? Any takers?

There have been requests to publish the project as a book, what do you think?

Any correspondence regarding the Hi-Res computer to:-

Teddy Technical
Daventry Design Dungeon
40/42 Oxford Street
Daventry
Northants
NN1 4AD

P.S. Please do not send money, cheques, gold bullion etc. to Teddy as he doesn't have a bank account or money-box. No individual correspondence can be entered into and any answers are via the column only – Thank you. TT.

Fig. 6 Main Timing Chart

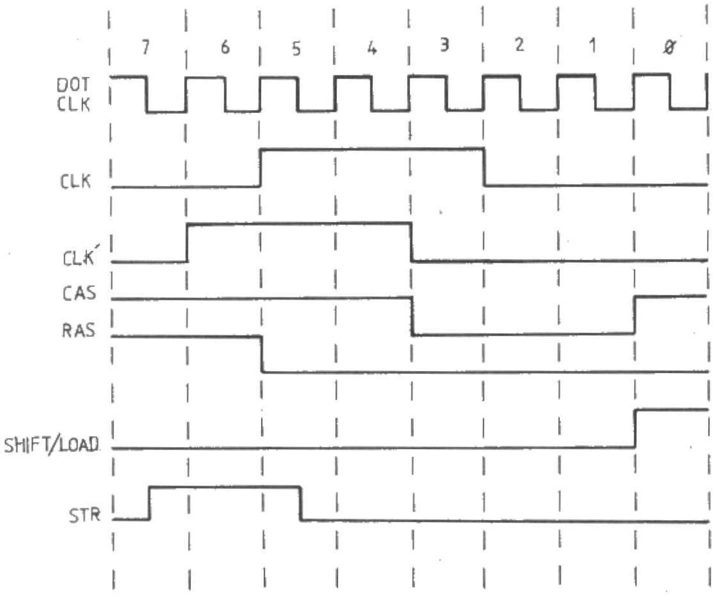


Fig 11

Keyboard Connector (on Hi-Res)	Keyboard Connector (See Fig. 10)
1 0V	
2 0V	GND
3 —12V	—12V
4 KBD	STROBE PULSE
5 KD0	B1
6 KD1	B2
7 KD2	B3
8 KD3	B4
9 KD4	B5
10 KD5	BIT 6 ALPHA
11 KD6	B7
12 KD7	NO CONNECTION
13	
14	
15	
16	
17	
18	
19 5V	
20 5V	+5

Fig. 7

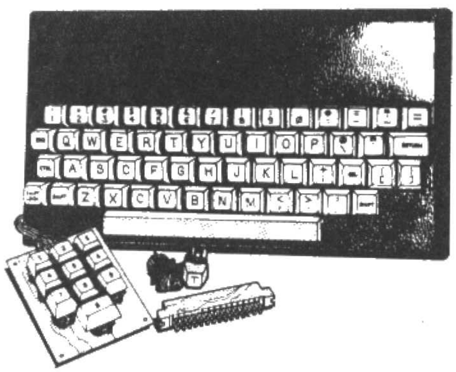
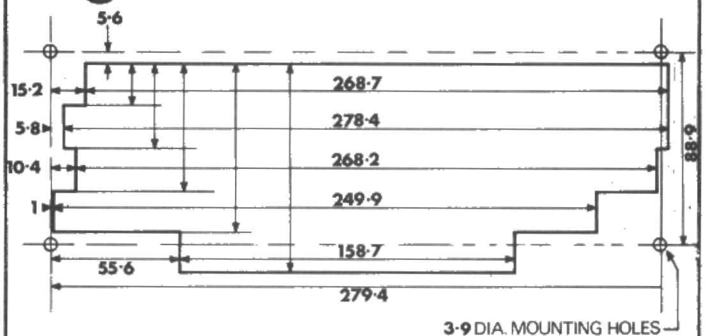
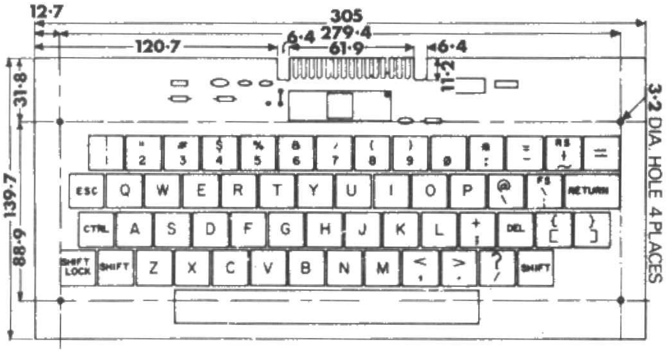


Fig. 9



SUGGESTED PANEL CUTOUT

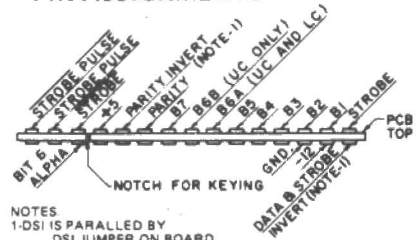
Fig. 8



KEYBOARD LAYOUT AND DIMENSIONS

Fig. 10

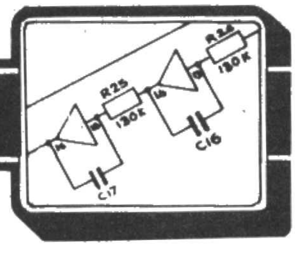
PIN ASSIGNMENTS



NOTES.
1-DSI IS PARALLED BY
DSI JUMPER ON BOARD

JUMPER	DATA AND STROBE OUTPUTS	PARITY OUTPUT
5	NEG LOGIC	EVEN
GND.	POS LOGIC	ODD

2-B6A PROVIDES SELECTABLE UPPER AND LOWER CASE
B6B PROVIDES UPPER CASE ONLY
BIT 6 ALPHA IS THE BIT 6 OUTPUT WHEN ALPHA LOCK OPTION IS USED.
3-STROBE PULSE WIDTH 1 MS WITH STD. VALUES
4-DRIVE CAPABILITY IS ONE TTL UNIT LOAD



Adding A Keyboard

For those who have built and commissioned the system so far, using a serial terminal, this may be superfluous because we have decided to produce two alternative monitor programs. The first will support a serial terminal – plus the Hi-Res Board (i.e. the program listing on the terminal and the graphics display on a colour monitor) and the second supports the Hi-Res board alone, using the parallel keyboard interface, and uses one page to support the operating system and the other for graphics.

This was done to satisfy demand from the Teddy Technical correspondents (this is becoming the world's first reader modified published computer design!).

So, if you have a terminal and are happy with it – it is not necessary to junk it (which is just as well, because I was wondering what to do with mine!).

The terminal monitor program will be known as HT-Bug and the keyboard version as HI-Bug. S-Bug can be used if the system is cassette based but some useful subroutines contained in the new Proms will not be available.

Preparing The Keyboard

For the following information we are endebted to Carter Keyboards and Watford Electronics.

As previously stated, the preferred keyboard is Carter type 756MF (which is the strengthened version).

A plastic case, precut to take the keyboard (Car/20) may be used (Fig. 7) or for real D.I.Y. masochists one of the many keyboard boxes that are readily available may be cut out to the pattern shown in Figs. 8 and 9.

Near the large chip on the keyboard is a link labelled D.S.I. (data strobe invert). This should be connected to "+" which selects normal data and strobe outputs.

The keyboard also has an alpha-lock option, which we will implement. When this is selected, only upper case characters will be output, and when deselected upper and lower cases are available.

This can be very useful when using the monitor, or other utilities, which do not recognise lower case characters and, thus, relieve the tedium of continually having to hold down the shift key every time a letter is required.

All that we need to do to accomplish this miracle is a 74L00 and 4.7K resistor!

These should be fitted to the board. in the appropriate labelled places.

Two jumpers will be found at the rear of the shift-lock key. These will normally be connected between "com" and "shift" and should be removed and reconnected between "com" and "alpha". This completes the modifications. Alpha-lock is selected when the shift lock is "down".

All that remains to complete the connection is to make up the ribbon cable as per Fig. 11.

Next Month — The new System Monitors and using the board.

If you're still staggering through the computer jungle and not getting sensible answers to your questions, we have some good news:

Now you can turn to **professional** people who are capable of giving you sound advice on practically all aspects of popular computing.

They all have one thing in common:

They are **COMPUTERS FOR ALL** dealers.

A Computers for All dealer is different from the normal Computer retailer.

Not surprisingly he won't try to sell you things like cameras or cosmetics; stationery or sealing wax.

He will, however, be capable of answering sensibly almost any question you have on computers and computing, and have readily available a wide range of popular computers, hardware, software, books, and peripherals.

So why not call in at your local **COMPUTERS FOR ALL** dealer today? He can lead you in the direction you want to travel.

The shops where people matter:

Akhter Instruments, Unit 19, Arlinghyde Estates, South Road, Harlow, Essex. 0279 412639. **Anirog Computers,** 26 Balcombe Gardens, Horley, Surrey. 02934 6083. **Audio & Computer Centre,** 7 Peter Street, St. Helier, Jersey, Channel Islands. 0534 74000. **Automation Services (S. Wales),** 3 Wermeys Road, Penysfai, Bridgend, S. Wales. 0656 720959. **Bits & Bytes,** 44 Fore Street, Ilfracombe, N. Devon EX34 9JD. 0271 62801. **Carlton Computers,** 4 Swanstons Road, Great Yarmouth, Norfolk NR30 3NQ. 0493 58898. **Computasolve,** 8 Central Parade, St. Marks Hill, Surbiton, Surrey. 01-390 5135. **Computer Corner,** 15 The Broadway, Wickford, Essex. 03744 3710. **Computer Systems (Torbay),** Pump Street, Brixham, Devon. 08045 6565/6. **Computers for All,** 72 North Street, Romford, Essex. 0708 752862. **Crystal Computers,** 209 Union Street, Torquay, Devon. 0803 22699. **D. V. Martin Ltd.,** 13 Bridge Street, Belfast, N. Ireland BT1 1LT. 0232 226434. **Dan Evans (Barry) Ltd.,** 81 Holton Road, Barry, South Glamorgan. 0446 734242. **Emprise Ltd.,** 58 East Street, Colchester, Essex. 0206 870353. **Emprise Ltd.,** 3a Baddow Road, Chelmsford, Essex. 0245 356834. **Fal-Soft Computers,** 8 St. George's Arcade, Falmouth, Cornwall. 0326 314663. **Home Computer Centre (Ross Records),** 13 Kingston Road, Portsmouth, Hampshire. 0703 819515. **Impulse Micro Systems Ltd.,** 6 Central Chambers, Cooks Alley, Wood Street, Stratford-upon-Avon. 0789 295819. **Jade Computers,** Coombend, Radstock, Bath, Avon. 0761 32570. **Kelly's Computermarket,** 227 Dartmouth Road, Sydenham, London SE26 4QY. 01-699 4399/6202. **Kenneth Ward Computers,** KF&R House, London Road, Ascot, Berkshire. 0990 22275. **Medway Computers,** 141 New Road, Chatham, Kent. 0634 826080. **Mercator Computer Systems,** 3 Whiteladies Road, Clifton, Bristol. 0272 731079. **Mobile Micros,** 2 Castle Street, Thornbury, Bristol. 0454 418383. **Steve's Computer Co. Ltd.,** Castle Arcade, Cardiff. 0222 41905. **The Computer Centre (BMS) Ltd.,** 37d&e, Robertson Street, Hastings, East Sussex. 0424 439190. **Twillstar Computers,** 17 Regina Road, Southall, Middlesex. 01-574 5271. **Weytech,** 20 St. Edmunds Street, Weymouth, Dorset. 03057 79881.

COMPUTERS FOR ALL